Università degli Studi di Padova
Dipartimento di Scienze Statistiche
Corso di Laurea Magistrale in
Scienze Statistiche

RELAZIONE FINALE

Un approccio bayesiano non parametrico per l'analisi di dati funzionali con dipendenza spazio temporale

Relatore Prof. Bruno Scarpa
Dipartimento di Scienze Statistiche

Alberto Grassi
Matricola N. 1221264

Anno Accademico 2021/2022

A tutte le persone che hanno
condiviso con me questi anni

Indice

Elenco delle figure

Elenco delle tabelle

Introduzione

I metodi bayesiani non parametrici stanno diventando un vero e proprio paradigma nell'analisi di dati complessi grazie alla loro estrema flessibilità. A differenza però dei metodi definiti "*black box*" essi mantengono la specificazione di una struttura completamente probabilistica. L'approccio bayesiano ha il vantaggio di lavorare in ogni fase dell'analisi con l'intera distribuzione della quantità di interesse, gestendo in modo completo e rigoroso tutta l'incertezza coinvolta nella stima. Esso consente di introdurre tramite la specificazione della distribuzione a priori informazioni o conoscenze pregresse, che possono essere più o meno vaghe, e modificare sulla base dell'evidenza empirica questa conoscenza in un processo che viene definito apprendimento bayesiano. In contesti complessi come quello in cui ci si pone, ovvero l'analisi di dati funzionali, si dispone raramente di un modello parametrico sufficientemente parsimonioso e flessibile che si adatti bene ai dati, e da questa motivazione nasce l'utilizzo dei metodi bayesiani non parametrici.

Il contesto applicativo trattato per l'analisi di dati funzionali è quello delle curve di mortalità, in particolare della distribuzione dei decessi per età, che rappresenta uno degli strumenti più importanti in ambito demografico per lo studio della mortalità come indice di progresso e salute di una popolazione. Nel Capitolo 1 viene proposta una breve introduzione al tipo di dati analizzati, alle altre quantità presenti nella tavola di mortalità e ad alcuni modelli parametrici noti in letteratura in questo ambito.

Modellando curve di mortalità si entra nel più ampio tema dei dati funzionali, i quali presentano caratteristiche peculiari rispetto ai classici dati che si è soliti analizzare. Questi dati infatti giacciono in uno spazio funzionale

e sono definiti su infiniti punti. Per la loro analisi si rende dunque necessario il ricorso a strumenti matematici appositi. Nel Capitolo 2 vengono introdotti a tale scopo i processi stocastici come distribuzioni su spazi funzionali. Questi sono utilizzati per la costruzione del modello basato sul processo di Dirichlet, uno degli strumenti più utilizzati in ambito bayesiano non parametrico. Nel Capitolo 3 viene presentata un'estensione di questo modello basata sul processo *Probit Stick-Breaking* per tener conto dell'informazione spaziale e temporale associata alle curve. Nel Capitolo 4 vengono applicati i modelli bayesiani non parametrici proposti alle curve di mortalità degli stati europei. Lo scopo è quello di riuscire a cogliere l'evoluzione temporale e spaziale di queste curve e ottenere un loro raggruppamento grazie alle proprietà dei modelli utilizzati. Questo risulta rilevante per i demografi perché permette di avere notevoli informazioni che contribuiscono alla comprensione delle dinamiche della popolazione.

Capitolo 1

Analisi di dati funzionali

1.1 Caratteristiche dei dati

In diverse applicazioni l'interesse principale consiste nello studiare la variabilità di funzioni casuali. In questi casi la variabile casuale oggetto di analisi non è scalare o vettoriale ma giace in uno spazio funzionale ed è definita su infiniti punti. Per studiare l'eterogeneità tra le osservazioni funzionali si considera un modello che ha la seguente forma

$$y_i(t) = f_i(t) + \varepsilon_i(t), \qquad \text{con } i = 1, \ldots, n$$

dove $y_i(t)$ è l'osservazione soggetta ad errore della funzione casuale $f_i(t)$, dipendente dall'indice t, e n rappresenta il numero di osservazioni funzionali. Diversi modelli di tipo parametrico o non parametrico differiscono per la specificazione dei termini $f_i(t)$ e $\varepsilon_i(t)$. Nella pratica tuttavia, è possibile raccogliere misurazioni di queste funzioni solo su un insieme finito di punti e dell'intero spazio funzionale si osserva una porzione finita, determinata dai punti $t_1, \ldots, t_T \in \mathcal{T}$. Tali punti possono essere ad esempio istanti temporali, possibilmente equispaziati, se si ipotizza una funziona casuale del tempo o età dell'individuo se si ipotizza ad una funzione casuale dell'età. Più in generale questi punti rappresentano specifici istanti di una qualsiasi variabile da cui dipende la funzione $f_i : \mathcal{T} \to \mathbb{R}$. Si osservi che i punti in cui la funzione viene

misurata possono anche differire da un'osservazione all'altra, in questi casi si avrebbe un $t_{ij} \in \mathcal{T}$ con $j = 1, \dots, T_i$ e $i = 1, \dots, n$.

L'approccio classico comunemente utilizzato nell'analisi funzionale (*Functional Data Analysis, FDA*) consiste nel lisciamento preliminare delle osservazioni tramite l'utilizzo di una qualche tecnica non parametrica. I metodi che utilizzano basi di funzioni risultano a tale scopo molto popolari anche grazie alla loro efficienza computazionale. Un sistema di funzioni di base è un insieme di funzioni note $\phi(t)$ linearmente indipendenti, le quali, tramite combinazione lineare composta da un numero di termini M sufficientemente elevato, possono approssimare arbitrariamente bene qualsiasi funzione. Pertanto è possibile approssimare una generica funzione tramite la seguente combinazione lineare

$$f_i(t) = \sum_{m=1}^{M} \beta_{im}\phi_m(t)$$

dove $\phi_m(t)$ rappresenta l'm-esimo elemento della base di funzioni. Un esempio è la base polinomiale i cui elementi sono $\{1, t, t^2, t^3, \dots\}$, ma si possono usare basi di funzioni più efficienti come le serie di Fourier o le basi indotte dalle *splines*. Queste ultime rappresentano funzioni polinomiali a tratti e necessitano per questo di un grado inferiore rispetto ad un'ipotetica base polinomiale altrettanto flessibile. I lisciatori basati su queste tecniche risultano particolarmente efficienti in quanto, fissata la base di funzioni, il problema della stima del vettore dei parametri $\beta_i = (\beta_{i1}, \dots, \beta_{iM})$ si traduce in un problema di regressione lineare per il quale si dispone di una soluzione in forma chiusa. Ad esempio, utilizzando i minimi quadrati si ottiene $\hat{\beta}_i = (\Phi^\top\Phi)^{-1}\Phi^\top y_i$, con $\Phi = (\phi_1, \dots, \phi_M)$ ma è possibile adoperare anche metodi di regolarizzazione con penalizzazione L_2 come lo stimatore *Ridge*, ottenendo $\tilde{\beta}_i = (\Phi^\top\Phi + \lambda I)^{-1}\Phi^\top y_i$, dove λ è un parametro di regolarizzazione che può essere scelto insieme al numero M tramite convalida incrociata.

Un altro problema classico della *FDA* è il problema non supervisionato del raggruppamento funzionale (Jacques e Preda, 2014). A tale scopo è possibile applicare gli usuali metodi di raggruppamento sia ai dati grezzi osservati sia ai dati lisciati con uno dei metodi precedentemente accennati. In entrambi i casi risulta necessario definire una misura di dissimilarità fra

curve, che può avvenire anche dal confronto delle stime dei coefficienti β_i qualora venga effettuato un lisciamento preliminare basato su una base di funzioni. Con dati di tipo funzionale il metodo di raggruppamento partizionale maggiormente utilizzato è il *Partitioning Around Medoids (PAM)*, il quale presenta diversi vantaggi rispetto al più noto k-medie. Primo fra tutti, l'algoritmo *PAM* presenta il vantaggio di rispecchiare nei medoidi la struttura liscia delle osservazioni, eventualmente ottenuta tramite lisciatura preliminare, e questo risulta utile per l'interpretazione dei gruppi individuati dall'algoritmo. Ciascun medoide infatti è formato dall'osservazione più centrale del gruppo, selezionata come rappresentativa di tutte le altre, e non viene ottenuto come media di tutte le osservazioni allocate nel gruppo come avviene invece per il calcolo dei centroidi utilizzati nel k-medie. Il secondo vantaggio dell'algoritmo *PAM* è che risulta meno sensibile del k-medie alla presenza di osservazioni anomale, tipicamente molto influenti nei metodi che raggruppano tutte le osservazioni in uno ed un solo gruppo e dove il numero di gruppi viene fissato a priori.

Sia che lo scopo sia quello di modellazione che quello di raggruppamento, in entrambi i casi nell'analisi funzionale classica si segue solitamente un procedimento di stima in due stadi con un lisciamento preliminare dei dati osservati. Nei capitoli successivi di questo lavoro viene presentato un approccio alternativo che permette di definire un modello probabilistico tale per cui una forma di raggruppamento è intrinseca nel modello stesso. Grazie alla natura discreta del processo di Dirichlet infatti, è possibile ottenere un raggruppamento delle osservazioni guardando quelle che condividono gli stessi atomi. In questo modo non serve specificare a priori il numero di gruppi presenti e si lascia che siano i dati ad informare sul numero di gruppi necessari. Inoltre, tale approccio non necessita nemmeno di un lisciamento preliminare delle osservazioni in quanto, con un'opportuna specificazione della distribuzione di base, gli stessi atomi funzionali estratti dal modello saranno lisci e potranno essere utilizzati come lisciatori per le curve allocate nel rispettivo gruppo.

Riferimenti testuali per l'analisi funzionale classica dal punto di vista frequentista sono Ramsay e Silverman (2005) per una trattazione teorica,

mentre Ramsay, Hooker e Graves (2009) per maggior enfasi sugli aspetti di implementazione software.

1.2 Contesto applicativo

1.2.1 Le curve di mortalità

Come esempio di dati funzionali si vuole considerare un contesto in ambito demografico analizzando la curva della distribuzione delle morti per età, che rappresenta una delle quantità riportate nella tavola di mortalità. Questa curva possiede una caratteristica forma multimodale con due o tre massimi locali, riferiti a specifiche componenti della mortalità sulle quali molti modelli di tipo parametrico basano i propri assunti inerenti alla forma funzionale.

La tavola di mortalità rappresenta uno dei più importanti strumenti utilizzati in ambito demografico per l'analisi statistica dei decessi e della loro incidenza per età e per genere. Essa permette di misurare il progresso di una popolazione, indicando il raggiungimento dell'obiettivo della longevità, e si presta perfettamente a confronti tra gruppi diversi, mettendone in luce differenze e analogie (Preston, Heuveline e Guillot, 2001).

Si immagini di avere a disposizione i tassi di mortalità specifici per età, indicati con m_x, ottenuti come rapporto tra le stime del numero di decessi e degli esposti al rischio per età e per anno. Vengono elencate nel seguito le quantità necessarie per la costruzione della tavola nel caso si considerino intervalli di età unitari da 0 a 110 anni e un intervallo finale aperto $[110, \infty)$ (Wilmoth et al., 2019):

- a_x: numero medio di anni vissuti nell'intervallo tra l'età x e $x+1$ per coloro che muoiono nell'intervallo, per questa quantità si assume che:

$$\begin{cases} a_x = \frac{1}{2} & \text{per } x = 1, 2, \ldots, 109 \\ {}_\infty a_{110} = \frac{1}{{}_\infty m_{110}} & \text{per } x = 110 \end{cases}$$

  mentre per l'età 0 si possono seguire le formule di *Coale-Demeny* per le quali si ha che a_0 è fissato pari a 0.33 per i maschi e a 0.35 per le

femmine se $m_0 \geq 0.107$, mentre è pari a $0.045 + 2.684m_0$ per i maschi e a $0.053 + 2.800m_0$ per le femmine se $m_0 < 0.107$;

- q_x: probabilità di morire tra l'età x e l'età $x+1$:

$$\begin{cases} q_x = \frac{m_x}{1+(1-a_x)m_x} & \text{per } x = 0, 1, 2, \ldots, 109 \\ {}_\infty q_{110} = 1 & \text{per } x = 110; \end{cases}$$

- p_x: probabilità di sopravvivere tra l'età x e l'età $x+1$, data dal complemento ad uno di q_x:

$$p_x = 1 - q_x;$$

- l_x: numero di sopravvissuti all'età $x > 0$ su una popolazione fittizia di $l_0 = 10^5$ persone:

$$l_x = l_0 \prod_{i=0}^{x-1} p_i;$$

- d_x: distribuzione delle morti per età:

$$\begin{cases} d_x = l_x q_x & \text{per } x = 0, 1, 2, \ldots, 109 \\ {}_\infty d_{110} = l_{110} & \text{per } x = 110. \end{cases}$$

La scelta di modellare le funzioni d_x rappresentanti la distribuzione delle morti per età dipende dal fatto che queste curve sono di più facile interpretazione rispetto alle altre e a partire da queste è più facile identificare alcune caratteristiche della mortalità come ad esempio l'età modale, la compressione attorno all'età modale e l'intensità di morti infantili.

Si noti che le funzioni d_x costituiscono una misura direttamente comparabile per il confronto della mortalità in diversi periodi o in diverse aree geografiche essendo costruite su una popolazione fittizia di centomila persone. In questo modo si corregge la distorsione che la composizione della popolazione avrebbe sul numero di morti. Utilizzando la distribuzione osservata dei decessi per fasce di età sulla popolazione reale di riferimento si potrebbero

registrare infatti differenze in termini di mortalità semplicemente dovute alla diversa composizione delle rispettive popolazioni.

1.2.2 Alcuni modelli di mortalità parametrici

Nel tempo sono state pubblicate diverse teorie sulle componenti che costituiscono la mortalità, accompagnate dal tentativo di riconoscerle a partire dalla rappresentazione della distribuzione dei decessi per età. La più famosa tra tutte fu introdotta da Lexis (1879), che divise la distribuzione dei decessi per età in tre componenti: infantile, prematura e normale. Secondo Lexis la prima componente inizia all'età 0 e finisce in corrispondenza del minimo della curva tra i 10 e i 12 anni. Per determinare l'area della mortalità adulta, Lexis considera la forma della distribuzione delle morti per età dall'età modale fino all'ultima età considerata e la capovolge verso sinistra ottenendo un'area simmetrica. Le morti che ricadono sotto quest'area sono considerate normali in quanto seguono la legge degli errori accidentali e riflettono la naturale durata della vita. Per esclusione la mortalità prematura rappresenta la regione di transizione dalla mortalità infantile a quella adulta ed è costituita da tutte le morti che avvengono al di fuori dell'area simmetrica precedentemente identificata.

Pearson (1897), partendo dagli studi svolti da Lexis, suddivise ulteriormente la curva di mortalità in cinque funzioni con un certo grado di asimmetria, introducendo un punto di vista più statistico e facendo una distinzione fra mortalità accidentale e prematura. In particolare egli propose una distribuzione esponenziale ed una fortemente asimmetrica per le due componenti della mortalità infantile, una distribuzione gaussiana centrata attorno all'età di 25 anni per la mortalità accidentale ed un'altra centrata attorno all'età di 40 anni per quella prematura e per finire una distribuzione con asimmetria a sinistra per la mortalità adulta.

Rappresentare con delle densità le diverse componenti della mortalità sfrutta l'idea che la distribuzione delle morti per età, moltiplicata per un fattore di scala di 10^{-5}, è una funzione di densità e può essere modellata quindi tramite una mistura di distribuzioni. Lavori più recenti in questa direzione

propongono dei modelli a mistura finita per approssimare e modellare in modo parsimonioso la curva di mortalità. In Mazzuco, Scarpa e Zanotto (2018) viene proposto un modello basato su una mistura di una distribuzione *Half Normal* per la mortalità infantile e una *Skew Normal* bimodale per la restante parte della curva, ottenendo un modello con soli sei parametri e sufficientemente flessibile. In Zanotto, Canudas-Romo e Mazzuco (2020) viene sostituita la *Skew Normal* bimodale con due *Skew Normal*, una per la mortalità accidentale e prematura insieme e l'altra per la mortalità adulta, ottenendo un modello con otto parametri formato dalla mistura di tre distribuzioni (Figura 1.1).

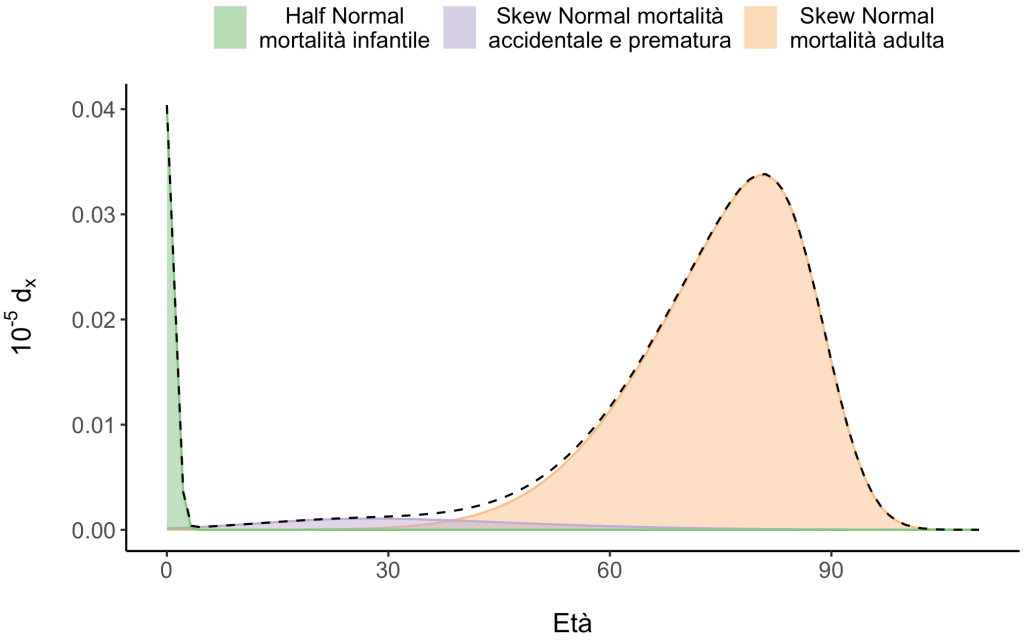


Figura 1.1: Rappresentazione delle tre componenti del modello mistura proposto da Zanotto, Canudas-Romo e Mazzuco (2020) per la distribuzione delle morti per età.

Si introduce nel seguito un classico modello parametrico noto in letteratura che servirà nel Capitolo 3 per la costruzione di una ragionevole media a priori del processo gaussiano, utilizzato come misura di base del modello bayesiano non parametrico, e un modello bayesiano parametrico che utilizza una passeggiata casuale per modellare in modo parsimonioso l'evoluzione

dinamica nel tempo e nello spazio delle curve di mortalità (Aliverti, Mazzuco e Scarpa, 2021).

Modello di Siler

Un modello particolarmente noto in letteratura fu quello proposto da Siler (1979). Questo modello rappresenta una generalizzazione di quello di Gompertz-Makeham (Gompertz, 1825; Makeham, 1860) e definisce il tasso di mortalità m_x all'età x come

$$m_x = a_1 e^{-b_1 x} + a_2 + a_3 e^{b_3 x}$$

dove a_1 rappresenta l'intensità della mortalità infantile e b_1 il tasso del suo declino mentre a_2, a_3 e b_3 sono i parametri del modello di Gompertz-Makeham, e descrivono rispettivamente un rischio costante nel tempo, la forma e il tasso della mortalità adulta. Si noti che il più semplice modello di Gompertz-Makeham risulta annidato in questo e infatti può essere ottenuto ponendo $a_1 = 0$. Il modello di Siler ha quindi cinque parametri che tengono conto del rischio di morire all'inizio della propria vita, del rischio costante durante tutta la durata della vita e di un incremento del rischio dovuto alla senescenza.

Modello DYSM

Il modello proposto da Aliverti, Mazzuco e Scarpa (2021) utilizza una mistura di tre distribuzioni asimmetriche per modellare la curva di mortalità in modo parsimonioso. La scelta delle distribuzioni che compongono la mistura sfrutta la nota forma caratteristica di questa curva e il fatto che può essere intesa come una funzione di densità che assume unicamente valori positivi ed integra ad uno. In particolare per la mortalità infantile viene scelta una distribuzione di Dirac, che risulta particolarmente adeguata per i paesi sviluppati nei quali la mortalità infantile si verifica entro il primo anno e può essere modellata da un singolo punto di massa in zero. In analogia con quanto proposto da Zanotto, Canudas-Romo e Mazzuco (2020), tale distribuzione può essere sostituita con una *Half Normal* per descrivere la mortalità infantile

nei paesi dove quest'ultima è ancora elevata. Per la mortalità accidentale e prematura viene scelta una distribuzione gaussiana mentre per la mortalità adulta una distribuzione *Skew Normal*. Questa scelta porta alla seguente densità per la distribuzione dei morti per età nel paese j e nell'anno t:

$$f(x;\boldsymbol{\theta}_{jt}) = \pi_{0\,jt}\delta_0 + \pi_{1\,jt}\phi\Big(\frac{x-\mu_{jt}}{\sigma_{jt}}\Big) + \pi_{2\,jt}\frac{2}{\omega_{jt}}\phi\Big(\frac{x-\xi_{jt}}{\omega_{jt}}\Big)\Phi\Big(\alpha_{jt}\frac{x-\xi_{jt}}{\omega_{jt}}\Big)$$

caratterizzata da sette parametri $\boldsymbol{\theta}_{jt} = (\pi_{1\,jt}, \pi_{2\,jt}, \mu_{jt}, \sigma_{jt}, \xi_{jt}, \omega_{jt}, \alpha_{jt})$, dove δ_0 indica la Dirac in zero, $\phi(\cdot)$ e $\Phi(\cdot)$ le funzioni di densità e di ripartizione di una normale standard. Si noti che per ottenere una densità i pesi della mistura devono sommare ad uno e dunque $\pi_{0\,jt} = 1 - \pi_{1\,jt} - \pi_{2\,jt}$ non è un parametro del modello.

Rispetto agli altri modelli, il DYSM modella l'evoluzione della curva di mortalità introducendo una dinamica evolutiva al vettore di parametri $\boldsymbol{\theta}_{jt}$. In particolare viene specificato un *trend* stocastico nel tempo mediante l'utilizzo di un *random walk* con *drift* e innovazioni gaussiane. Di fatto è possibile vedere questo modello come un modello *state-space* con equazione di transizione

$$(\tilde{\theta}_{jtk}|\tilde{\theta}_{jt-1k}, \beta_{jk}, \eta^2_{jk}) \sim N(\beta_{jk} + \tilde{\theta}_{jt-1k}, \eta^2_{jk}) \quad \text{per } k = 1, \ldots, 7,$$

dove il vettore $\tilde{\boldsymbol{\theta}}_{jt}$ rappresenta una riparametrizzazione del vettore $\boldsymbol{\theta}_{jt}$ originale in cui per i due pesi della mistura $\pi_{1\,jt}$ e $\pi_{2\,jt}$ viene applicata una trasformata a *logit* cumulati mentre per i parametri relativi alla varianza σ_{jt} e ω_{jt} una trasformata logaritmica. Questa riparametrizzazione ha lo scopo di mappare lo spazio parametrico in $\mathbb{R}^7$, che risulta di più facile gestione essendo non vincolato. Il parametro β_{jk} misura in questo caso la differenza attesa fra due valori consecutivi di $\tilde{\theta}_{jtk}$.

Seguendo un approccio bayesiano viene specificata una distribuzione a priori comune per i parametri β_{jk} e η^2_{jk} dei diversi paesi:

$$\beta_{jk} \sim N(m_{\beta_k}, s_{\beta_k}) \quad \eta^2_{jk} \sim IGa(a_k, b_k)$$

dove $m_{\beta_k}, s_{\beta_k}, a_k$ e b_k sono iperparametri fissati. Si osservi che l'utilizzo di una distribuzione a priori comune consente di specificare un modello gerarchico che sfrutta l'informazione proveniente da tutte le curve dislocate nello spazio per la stima dei parametri delle curve dei singoli paesi. Questa presa in prestito d'informazione fra i diversi paesi è giustificata dal fatto che nella maggior parte degli stati sviluppati si osservano tendenze simili e questa abbondanza d'informazione può essere opportunamente inclusa nel modello per migliorare la qualità delle stime. Altra caratteristica che segue dall'utilizzo di un modello gerarchico è la regolarizzazione implicita che si ottiene delle stime dei parametri dei singoli paesi. Se in generale con modelli gerarchici si osserva uno schiacciamento delle stime attorno alla media generale, con un modello bayesiano si ottiene uno schiacciamento delle distribuzioni a posteriori attorno alla distribuzione a priori, andando a penalizzare scenari particolari diversi dagli altri.

1.3 Scelte di modellazione

Visti i diversi approcci parametrici che si possono seguire per la modellazione delle curve di mortalità, si intende proseguire il lavoro con la scelta di un approccio più flessibile e più generalizzabile al più ampio contesto di dati funzionali. In questo caso verranno utilizzate le informazioni disponibili sulla forma funzionale dei dati da analizzare per la sola definizione di una ragionevole media a priori di un processo gaussiano, il quale deve avere un'opportuna matrice di covarianza che permetta di regolare il grado di concentrazione e lisciatura delle estrazioni. Il processo gaussiano viene utilizzato come misura di base di un modello bayesiano non parametrico, il quale riesce a garantire una grande flessibilità e fornisce un raggruppamento delle osservazioni in gruppi funzionali sintetizzati da atomi lisci.

Con dati funzionali di diversa natura si potrebbe quindi semplicemente definire una diversa media a priori del processo gaussiano lasciando invariata la struttura generale del modello e dell'algoritmo di stima.

Capitolo 2

Modello basato sul processo di Dirichlet funzionale

2.1 L'approccio bayesiano non parametrico

In contesti complessi si dispone raramente di un modello parametrico sufficientemente parsimonioso e flessibile che si adatti bene ai dati e, inoltre, le ipotesi di modellazione imposte dalla scelta di uno specifico modello parametrico risultano difficilmente verificabili in dimensioni elevate. Un aspetto attraente degli approcci non parametrici è quello di non imporre una particolare struttura matematica alla funzione che si vuole modellare, assumendo solo alcune condizioni di regolarità nel suo andamento mediante strumenti che offrono una maggiore flessibilità (Azzalini e Scarpa, 2009). I modelli non parametrici sono da intendersi in realtà come modelli con un numero infinito di parametri. Gli infiniti parametri conferiscono maggior flessibilità e rappresentano l'incertezza che si avrebbe nella specificazione di un modello completamente parametrico.

I classici metodi non parametrici, tuttavia, funzionano tendenzialmente male in contesti complessi a causa della maledizione della dimensionalità e della difficoltà di includere informazione parziale nel modello. La maledizione della dimensionalità viene risolta dai metodi bayesiani non parametrici attraverso la specificazione di opportune distribuzioni a priori che regolarizzano la stima

dei parametri e attraverso la centratura attorno ad un modello parametrico di base. L'introduzione per questi parametri di una distribuzione a priori centrata su un modello parametrico di base favorisce lo schiacciamento verso una forma parametrica più semplice. La presenza di informazione parziale invece, come ad esempio la presenza di dati mancanti o di dati censurati, può essere risolta in fase di stima tramite tecniche di *data augmentation* e *variable augmentation* che in contesti bayesiani risultano particolarmente utili e semplici da utilizzare. Queste motivazioni hanno reso i metodi bayesiani non parametrici vantaggiosi nelle applicazioni pratiche e in contesti di elevata sparsità, rendendoli un paradigma ampiamente utilizzato che ottiene i vantaggi chiave di una struttura probabilistica completamente basata su modelli, pur rimanendo altamente flessibili (Hjort et al., 2010).

2.2 Il processo di Dirichlet

2.2.1 Definizione del processo

Il processo di Dirichlet (*Dirichlet Process, DP*) è il processo stocastico prevalentemente utilizzato in ambito bayesiano non parametrico e rappresenta uno dei principali strumenti per la costruzione di misure di probabilità casuali. I processi stocastici possono essere considerati come distribuzioni su spazi di funzioni che nel caso del processo di Dirichlet sono misure di probabilità. Proprio per il fatto che descrive una distribuzione di probabilità sullo spazio delle misure di probabilità, il processo di Dirichlet può essere visto come una "distribuzione di distribuzioni", termine con cui si intende dire che ciascuna estrazione da esso è a sua volta una distribuzione, discreta, che non può essere descritta da un numero finito di parametri.

Per introdurre da un punto di vista formale il processo di Dirichlet si definisce G_0 una distribuzione nello spazio Ω e α un valore reale positivo. Una distribuzione casuale G segue un processo di Dirichlet con distribuzione di base G_0 e parametro di concentrazione α se per ogni partizione finita e

misurabile $\mathcal{A}_1, \ldots, \mathcal{A}_r$ di Ω si ha che

$$(G(\mathcal{A}_1), \ldots, G(\mathcal{A}_r)) \sim Dir(\alpha G_0(\mathcal{A}_1), \ldots, \alpha G_0(\mathcal{A}_r))$$

dove $Dir()$ indica la distribuzione di Dirichlet. In questo caso si dirà che $G \sim DP(\alpha, G_0)$. Il nome del DP è dovuto quindi al fatto che esso induce delle distribuzioni di Dirichlet finito-dimensionali quando i dati vengono raggruppati in un numero finito di gruppi.

Per chiarire il ruolo del parametro di concentrazione e della distribuzione di base nel processo di Dirichlet, si noti che dato $G \sim DP(\alpha, G_0)$ e un insieme $\mathcal{A}_j \in \{\mathcal{A}_1, \ldots, \mathcal{A}_r\}$, per le proprietà della distribuzione di Dirichlet e sfruttando il fatto che G_0 è una densità su Ω e $\mathcal{A}_1, \ldots, \mathcal{A}_r$ una partizione di Ω, si ha che $G(\mathcal{A}_j) \sim Beta\Big(\alpha G_0(\mathcal{A}_j), \alpha(1 - G_0(\mathcal{A}_j))\Big)$ con

$$\mathbb{E}(G(\mathcal{A}_j)) = G_0(\mathcal{A}_j)$$
$$Var(G(\mathcal{A}_j)) = \frac{G_0(\mathcal{A}_j)(1 - G_0(\mathcal{A}_j))}{1 + \alpha}.$$

La distribuzione di base G_0 rappresenta quindi la media del processo di Dirichlet mentre α regola la concentrazione del processo intorno alla media, essendo inversamente proporzionale alla varianza. Per fornire un'idea anche visiva dell'impatto del parametro di concentrazione sulle realizzazioni del processo di Dirichlet, in Figura 2.1 si mostra come all'aumentare di α le G estratte dal processo risultino più concentrate attorno a G_0 (Müller et al., 2015). Si osservi che per $\alpha \to \infty$ vale che $G(\mathcal{A}_j) \to G_0(\mathcal{A}_j)$, tuttavia, questo non implica che $G \to G_0$. La convergenza vale infatti solo puntualmente e la distribuzione casuale indotta dal processo di Dirichlet rimane discreta anche nel caso in cui G_0 non lo sia, come si può vedere dalle funzioni di ripartizione a gradini. In questo senso la distribuzione di base G_0 può essere interpretata come il miglior tentativo di specificazione parametrica del modello e il parametro di concentrazione α come la fiducia nei confronti di tale specificazione.

Aspetto interessante del processo di Dirichlet è che forma una famiglia di distribuzioni coniugata, infatti considerando $\eta_i \overset{iid}{\sim} G, i = 1, \ldots, n$ estrazioni

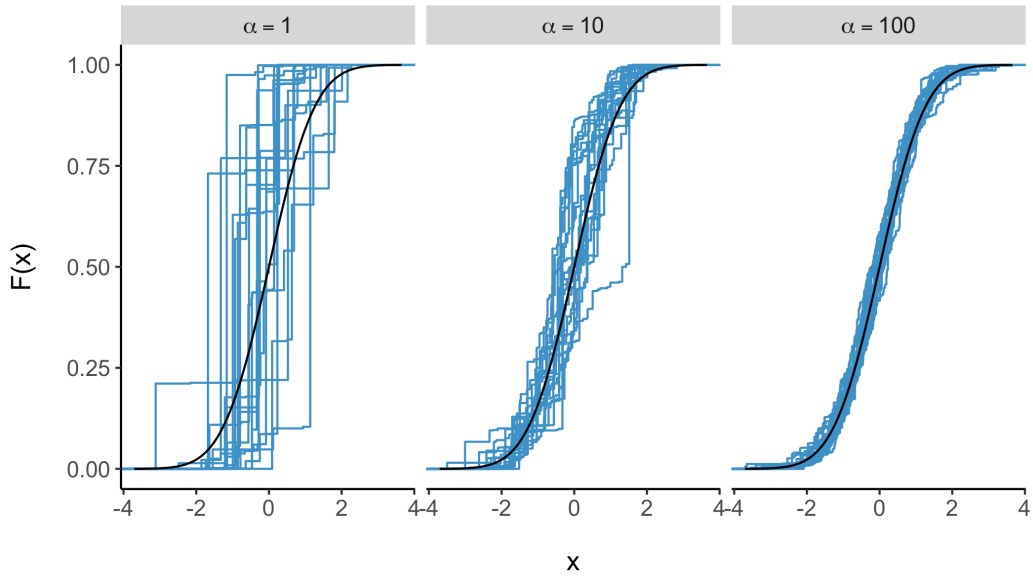


Figura 2.1: Rappresentazione delle funzioni di ripartizione empiriche di 20 campioni simulati da un $DP(\alpha, G_0)$, con G_0 normale *standard*, per $\alpha = 1, 10, 100$. Sovrapposta la funzione di ripartizione di G_0, dove $G_0 = \mathbb{E}(G)$.

indipendenti dalla distribuzione casuale G, la distribuzione a posteriori di G risulta essere

$$(G(\mathcal{A}_1), \ldots, G(\mathcal{A}_r))|(\eta_1, \ldots, \eta_n) \sim Dir(\alpha G_0(\mathcal{A}_1) + n_1, \ldots, \alpha G_0(\mathcal{A}_r) + n_r)$$

dove si è indicato con n_j il numero di η_i che appartengono ad $\mathcal{A}_j$, per $i = 1, \ldots, n$ e $j = 1, \ldots, r$. Questo equivale a specificare un processo di Dirichlet con parametro di concentrazione e distribuzione di base aggiornati nel seguente modo

$$G|(\eta_1, \ldots, \eta_n) \sim DP\left(\alpha + n,\ \frac{\alpha}{\alpha + n} G_0 + \frac{n}{\alpha + n} \frac{\sum_{i=1}^{n} \delta_{\eta_i}}{n}\right),$$

dove δ_{η_i} indica un punto di massa in η_i. Rispetto alla distribuzione a priori, la distribuzione a posteriori risulta essere più concentrata attorno alla nuova distribuzione di base in quanto il parametro di concentrazione cresce linearmente con n. La distribuzione di base a posteriori è una mistura tra la distribuzione di base della a priori, G_0, e la distribuzione empirica dei valori

osservati. La distribuzione empirica viene in questo modo compressa verso la media a priori. Il peso assegnato a G_0 è proporzionale al parametro di concentrazione della distribuzione a priori mentre il peso della distribuzione empirica è proporzionale al numero di osservazioni n, ovvero alla quantità di informazione disponibile dai dati. Si ottiene in questo modo una regolarizzazione bayesiana della stima della funzione di ripartizione, dove all'aumentare di n le estrazioni dal processo di Dirichlet condizionato ai dati osservati tenderanno ad assumere sempre più la forma dell'effettiva distribuzione empirica osservata.

2.2.2 Rappresentazione *stick-breaking*

Per comprendere la natura discreta della distribuzione casuale G viene presentata la rappresentazione *stick-breaking* (Gelman et al., 2015) del processo di Dirichlet. Assumere $G \sim GP(\alpha, G_0)$ è equivalente ad assumere

$$G = \sum_{h=1}^{\infty} w_h \delta_{\theta_h}, \quad \text{con } \theta_h \overset{iid}{\sim} G_0$$
$$w_h = v_h \prod_{r=1}^{h-1} (1 - v_r)$$
$$v_h \overset{iid}{\sim} Beta(1, \alpha)$$

dove i θ_h prendono il nome di atomi e i δ_{θ_h} sono punti di massa in θ_h. La distribuzione casuale G è dunque discreta in quanto assegna probabilità positiva ai singoli atomi.

Il nome *stick-breaking* deriva dal fatto che la misura di probabilità casuale viene costruita ricorsivamente "spezzando", secondo una qualche legge probabilistica, un ipotetico bastoncino di lunghezza unitaria ogni volta in proporzioni di v_h e $1 - v_h$ e poi continuando il procedimento in modo ricorsivo su $1 - v_h$. Il peso w_h assegnato all'atomo θ_h rappresenta la porzione v_h sulla lunghezza $\prod_{r=1}^{h-1}(1 - v_r)$ del bastoncino rimanente. Risulta intuitivo vedere come i pesi $w_1, w_2, \ldots$, detti *random weights*, diventino via via più piccoli. In particolare dalla conoscenza della distribuzione a priori dei v_h e quindi

dal fatto che $\mathbb{E}(v_h) = \frac{1}{1+\alpha}$, si può dimostrare come l'importanza dei *random weights* decresca geometricamente e il loro valore atteso (Rodriguez e Müller, 2013) sia pari a

$$\mathbb{E}(w_h) = \frac{1}{1+\alpha}\left(\frac{\alpha}{1+\alpha}\right)^{h-1}.$$

Il processo di Dirichlet è quindi interpretabile come una mistura infinita di variabili casuali provenienti da G_0. In generale tutti i modelli bayesiani non parametrici possono essere intesi come modelli mistura infinita e in questo senso rappresentano modelli con infiniti parametri. Il vantaggio di lavorare su una mistura infinita è quello di garantire sufficiente flessibilità oltre ad offrire la possibilità di approssimare tale mistura, anche in modo adattivo, utilizzando i primi N pesi della mistura. Per troncare la rappresentazione infinita è sufficiente imporre $v_N = 1$ e di conseguenza $w_{N+1} = w_{N+2} = \cdots = 0$. In questo modo ci si riconduce ad un modello mistura finita con N componenti che ben approssima il modello mistura infinita e da un punto di vista computazionale risulta di più semplice gestione. Quella che si ottiene infatti è una rappresentazione trattabile finito-dimensionale, che permette di usare tecniche di tipo *Markov Chain Monte Carlo* (MCMC) e che ben approssima il processo di Dirichlet oggetto di studio. Si raccomanda una scelta conservativa di N, in quanto determina il numero massimo di gruppi che si possono ottenere e non il numero effettivo.

Introdotta la rappresentazione *stick-breaking*, un altro modo per comprendere l'effetto del parametro α sulla concentrazione del processo è quello di osservare che tanto più α diventa piccolo, tanto più la distribuzione $Beta(1, \alpha)$ si schiaccia su valori prossimi a 1 (Figura 2.2), e di conseguenza tanto più grande sarà la porzione v_h eliminata, lasciando agli atomi successivi una porzione molto piccola. Avere un α piccolo implica quindi che la distribuzione G che si ottiene risulti una mistura effettiva di pochi atomi perché i w_h vanno velocemente a zero. La concentrazione del *DP* allora diminuisce nel senso che le G risultano molto diverse fra loro e distanti dalla distribuzione di base, che al contrario assume valori positivi su tutto il supporto dei possibili valori per gli atomi.

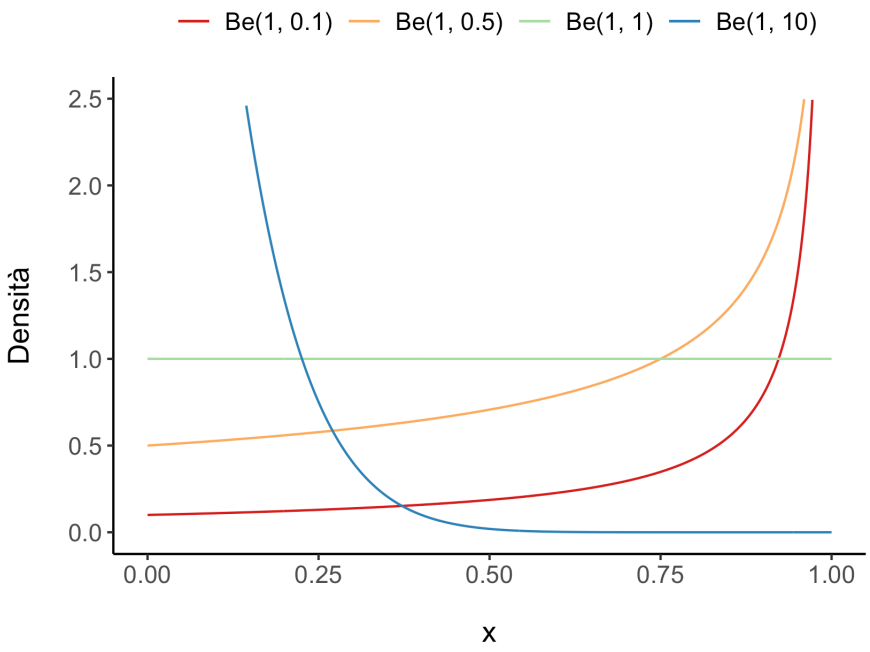


Figura 2.2: Densità della distribuzione $Beta(1, \alpha)$ per $\alpha = 0.1, 0.5, 1, 10$.

2.2.3 Raggruppamento indotto

Un'importante conseguenza già anticipata della natura discreta di G è che i valori estratti da un processo di Dirichlet hanno una probabilità non nulla di non essere tutti distinti fra loro. Questo definisce implicitamente un raggruppamento delle osservazioni, assegnando allo stesso gruppo tutte le osservazioni che condividono lo stesso atomo. Per comprendere il funzionamento è possibile ricostruire in maniera ricorsiva la distribuzione predittiva per una nuova realizzazione proveniente da $G \sim DP(\alpha, G_0)$. Partendo da

$$\begin{aligned} \eta_1 &\sim G_0 \\ G|\eta_1 &\sim DP\left(\alpha + 1,\ \frac{\alpha}{\alpha+1}G_0 + \frac{1}{\alpha+1}\delta_{\eta_1}\right) \\ \eta_2|G, \eta_1 &\sim G \end{aligned}$$

e marginalizzando rispetto a G, si ottiene la distribuzione della nuova osservazione condizionata alla precedente

$$\eta_2|\eta_1 \sim \frac{\alpha}{\alpha+1}G_0 + \frac{1}{\alpha+1}\delta_{\eta_1}.$$

Il valore di η_2 risulterà quindi pari al valore di η_1 con probabilità $\frac{1}{\alpha+1}$ mentre corrisponderà ad una nuova estrazione da G_0 con probabilità $\frac{\alpha}{\alpha+1}$. Generalizzando i passaggi precedenti è possibile ottenere la seguente funzione di probabilità predittiva (Blackwell e MacQueen, 1973) per il valore η_{n+1} avendo osservato n realizzazioni del *DP*

$$\eta_{n+1}|(\eta_1,\ldots,\eta_n) \sim \begin{cases} \delta_{\eta_j^*} & \text{con probabilità } \frac{n_j}{\alpha+n} \quad \text{per } j=1,\ldots,k \\ G_0 & \text{con probabilità } \frac{\alpha}{\alpha+n}, \end{cases}$$

dove si sono indicati con $\eta_1^*,\ldots,\eta_k^*$ i valori univoci di $\eta_1,\ldots,\eta_n$ e con $n_1,\ldots,n_k$ le rispettive frequenze assolute.

Questo processo di allocazione è noto anche come *Urna di Polya*. Intuitivamente si consideri un'urna che inizialmente contiene α palline nere e una colorata con un colore selezionato casualmente da G_0. Si effettuano estrazioni ripetute dall'urna. Quando viene estratta una pallina colorata, quest'ultima viene inserita nuovamente nell'urna insieme ad un'altra pallina dello stesso colore; mentre quando si estrae una pallina nera, quest'ultima viene reinserita insieme ad un'altra pallina di un colore selezionato casualmente da G_0 (Rodriguez e Müller, 2013). Un altro modo ampiamente utilizzato per descrivere questo processo di allocazione è il cosiddetto processo del ristorante cinese, secondo cui ci sono n clienti che vogliono sedersi in un ristorante con un numero infinito di tavoli, ognuno con capacità infinita. Il primo cliente si siede al primo tavolo e i clienti successivi occupano un tavolo già occupato o uno libero seguendo le probabilità indicate nella distribuzione predittiva, ottenendo alla fine una partizione degli n clienti in $m \leq n$ tavoli.

Alla luce di quanto mostrato si capisce quindi che il parametro di concentrazione α incide sul numero di gruppi individuati dal modello in quanto determina la probabilità di campionare un nuovo valore da G_0. Il nuovo valore

estratto, se diverso dai precedenti, andrà a formare infatti un nuovo gruppo. Si osservi che questo accade con probabilità 1 se G_0 è una distribuzione continua. A partire dalla distribuzione predittiva degli η_i, Antoniak (1974) ha ricavato che la distribuzione per il numero di gruppi $k \in \{1, \ldots, n\}$ è

$$p(k) = \frac{a_k \alpha^k}{A_n(\alpha)}$$

dove

$$\begin{aligned} A_n(\alpha) &= \alpha(\alpha+1)(\alpha+2)\cdots(\alpha+n-1) \\ &= a_n\alpha^n + a_{n-1}\alpha^{n-1} + \cdots + a_2\alpha^2 + a_1\alpha. \end{aligned}$$

Fissato il numero di osservazioni n, il numero di gruppi k, ovvero il numero di atomi distinti, ha valore atteso che dipende da α ed è pari a

$$\mathbb{E}(k) = \sum_{i=1}^{n} \frac{\alpha}{\alpha + i - 1} = \alpha(\psi(\alpha+n) - \psi(\alpha))$$

dove $\psi(\cdot)$ è la funzione digamma. Per $n \to \infty$, si ha che $\mathbb{E}(k) \simeq \alpha \log(\frac{n}{\alpha})$. In generale il numero di gruppi risulta inferiore rispetto al numero di osservazioni, con la tendenza ad accrescere ulteriormente i gruppi più numerosi. Per consentire ai dati stessi di informare sul numero di gruppi in essi presenti, spesso si assegna al parametro di concentrazione una distribuzione a priori. La distribuzione gamma è coniugata per il parametro α e dunque una scelta tipica non eccessivamente informativa è $\alpha \sim Ga(1, 1)$.

La proprietà di raggruppamento del processo di Dirichlet è stata ampiamente sfruttata negli ultimi anni, poiché presenta alcune caratteristiche che risultano interessanti nella pratica rispetto alle altre più note tecniche di raggruppamento accennate nel Capitolo 1. In particolare, il raggruppamento indotto dal processo di Dirichlet evita di assumere che tutti gli individui possano essere raggruppati in un numero fisso di gruppi, bensì assume che nella popolazione complessiva siano rappresentati infiniti gruppi e nel campione osservato un numero incognito finito. Ciascuna nuova osservazione ha infatti una probabilità non nulla di essere assegnata ad un nuovo gruppo non ancora

rappresentato nel campione.

Per concludere la discussione sul processo di Dirichlet si mette in luce un ulteriore vantaggio molto interessante della rappresentazione *stick-breaking*, e in particolare della sua versione troncata, ovvero la possibilità di generare nuove misure casuali sostituendo la distribuzione $Beta(1, \alpha)$ con altre distribuzioni casuali. Si vedrà nel Capitolo 3 che sostituendo la distribuzione Beta con un modello *probit* è possibile far dipendere la costruzione dei *random weights* da una serie di covariate specificando un opportuno modello di regressione.

2.3 Il processo di Dirichlet per dati funzionali

2.3.1 Specificazione del modello

Si introduce nel seguito un modello bayesiano non parametrico basato sul processo di Dirichlet funzionale (*Functional Dirichlet Process, FDP*). Rispetto a quanto presentato in precedenza, per ottenere un processo di Dirichlet per dati funzionali è sufficiente modificare la specificazione della distribuzione di base G_0, la quale deve riflettere la struttura, in questo caso funzionale, dei dati osservati. Nel contesto in analisi le osservazioni y_i sono realizzazioni di funzioni casuali contenenti una componente di errore che può differire da un'unità statistica all'altra. Si considera per queste osservazioni il seguente modello bayesiano non parametrico

$$
\begin{aligned}
y_i(t) &= \eta_i(t) + \varepsilon_i(t) \\
\eta_i &\sim G \\
G &\sim FDP(\alpha, G_0) \\
\varepsilon_i(t) &\sim N(0, \sigma^2)
\end{aligned}
$$

dove $y_i(t)$ rappresenta un'osservazione soggetta ad errore della funzione media η_i all'istante t. Più in generale i punti t in cui viene misurata la funzione possono essere diversi da osservazione a osservazione ma per l'applicazione alle curve di mortalità è sufficiente mantenere un t comune, che rappresenta le età o le classi di età alle quali viene calcolata la curva. Con $G \sim FDP(\alpha, G_0)$ si

intende che la distribuzione casuale G segue un processo di Dirichlet funzionale con distribuzione di base G_0 e parametro di concentrazione α. Anche il *FDP* è parametrizzato quindi da un parametro di concentrazione α, per il quale valgono le stesse considerazioni fatte per il *DP*, e da una distribuzione di base G_0, i cui atomi però sono in questo caso anch'essi funzionali. Ciascuna estrazione dal *FDP* rappresenta di conseguenza una distribuzione discreta sullo spazio delle funzioni $\Omega : \mathcal{T} \to \mathbb{R}$.

In letteratura esistono diversi modelli bayesiani non parametrici per l'analisi di dati funzionali basati sul processo di Dirichlet funzionale, e questi differiscono per la scelta della distribuzione di base. Una possibilità è quella di assumere che la distribuzione di base del processo di Dirichlet funzionale sia essa stessa un processo, e una scelta comune in questo senso è quella di definire G_0 come un processo gaussiano (*Gaussian Process, GP*) con funzione media μ e funzione di covarianza $\mathcal{K}$, che viene indicato con $GP(\mu, \mathcal{K})$. Un processo gaussiano è un oggetto infinito-dimensionale completamente specificato dalla sua funzione media e dalla sua funzione di covarianza. Esso consiste di fatto in una raccolta di variabili casuali tali per cui ciascun sottoinsieme di esse ha distribuzione normale multivariata. Si osserva che per poter operare concretamente, come detto per le osservazioni funzionali, questi processi vengono valutati in una griglia finita di punti e quindi nella pratica ci si riconduce a lavorare con distribuzioni normali multivariate, per le quali vengono specificati il vettore media e la matrice di covarianza.

Per il processo di Dirichlet funzionale con distribuzione di base un processo gaussiano la rappresentazione *stick-breaking* è la seguente

$$
\begin{aligned}
G &= \sum_{h=1}^{\infty} w_h \delta_{\Theta_h} \\
w_h = v_h \prod_{r=1}^{h-1} (1 - v_r), \ &\text{con } v_h \sim Beta(1, \alpha) \\
\Theta_h &\sim GP(\mu, \mathcal{K})
\end{aligned}
$$

dove gli atomi funzionali sono indicati con Θ_h per distinguerli da quelli del processo di Dirichlet non funzionale.

La funzione di covarianza, detta anche funzione nucleo, incide fortemente sulla forma e sulle proprietà del processo gaussiano. Scelta importante risulta quindi la specificazione di tale funzione in quanto le caratteristiche del processo gaussiano si riflettono sui Θ_h che definiscono i gruppi funzionali a cui le curve y_i, $i = 1, \ldots, n$, vengono allocate. Una scelta comune che induce processi lisci e differenziabili infinite volte è la funzione di covarianza esponenziale, che dipende da un solo parametro $\kappa > 0$ ed è definita come

$$\mathcal{K}(t, t'; \kappa) = \exp\{-\kappa(t - t')^2\}.$$

Si osserva che quando i valori t e t' sui quali viene calcata sono vicini, la funzione tende ad assumere valori vicino all'unità, mentre tende ad annullarsi all'aumentare della distanza tra i valori. La funzione di covarianza esponenziale è inoltre una funzione stazionaria in senso stretto, la cui distribuzione di probabilità congiunta dipende solo dalla distanza tra i valori forniti in ingresso e non dalla loro direzione e posizione, risultando quindi invariante a traslazioni e rotazioni nello spazio dei valori t e t'. Mantenendo invariate le caratteristiche, si può aggiungere alla funzione un secondo parametro con effetto moltiplicativo per regolarne la magnitudine. La funzione di covarianza che si utilizzerà nel seguito per il *GP* è una generalizzazione proposta da Dunson e Herring (2006) ed è definita nel seguente modo

$$\mathcal{K}(t, t'; \kappa_1, \kappa_2) = \kappa_1 \exp\{-\kappa_2(t - t')^2\} + \zeta\kappa_1 I(t = t'),$$

dove $\kappa_1 > 0$ controlla la magnitudine, $\kappa_2 > 0$ quanto deve essere liscia l'estrazione al variare della distanza fra t e t', e $\zeta > 0$ è un numero piccolo fissato che viene aggiunto lungo la diagonale principale della matrice per ragioni computazionali. La funzione di covarianza esponenziale si ottiene ponendo $\zeta = 0$.

Per completare la specificazione del modello basato sul *FDP* con distribuzione di base il processo gaussiano con funzione di covarianza esponenziale generalizzata, da un punto di vista completamente bayesiano si definiscono le

seguenti distribuzioni a priori per i parametri

$$\sigma^2 \sim IGa(a_\sigma, b_\sigma) \qquad \kappa_1 \sim Ga(a_{\kappa_1}, b_{\kappa_1})$$
$$\alpha \sim Ga(a_\alpha, b_\alpha) \qquad \kappa_2 \sim Ga(a_{\kappa_2}, b_{\kappa_2})$$

dove IGa indica la distribuzione Gamma Inversa. Questo è equivalente a specificare una distribuzione Gamma per il reciproco del parametro, ossia $\sigma^{-2} \sim Ga(a_\sigma, b_\sigma)$ con $\mathbb{E}(\sigma^{-2}) = a_\sigma/b_\sigma$ e $\mathbb{V}(\sigma^{-2}) = a_\sigma/b_\sigma^2$. Una scelta comune poco informativa per tutte queste distribuzioni a priori è quella di fissare entrambi gli iperparametri ad uno stesso valore piccolo, e positivo per i vincoli sullo spazio parametrico della distribuzione Gamma, in modo da ottenere distribuzioni con elevata varianza.

Come detto precedentemente l'approccio basato sui processi gaussiani non è il solo percorribile. Si cita l'approccio basato sulle *splines* come alternativa all'uso dei processi gaussiani come distribuzione di base del *FDP*. In questo caso ciascun atomo funzionale viene scritto come

$$\Theta_h = \sum_{l=1}^{L} \beta_{hl} b_l,$$

dove $\beta_h = (\beta_{h1}, \ldots, \beta_{hL})^\top$ sono i coefficienti delle *splines* per l'h-esimo atomo funzionale, mentre $\{b_l\}_{l=1}^L$ rappresenta una base di *splines*. L'utilizzo delle *splines* richiede tuttavia un'esplicita scelta dell'insieme delle funzioni di base, del loro grado, del numero e della posizione dei nodi, e questo rende il loro utilizzo meno immediato.

2.3.2 Algoritmo di stima

Qualunque sia la scelta della distribuzione di base G_0, le distribuzioni a posteriori dei parametri non sono trattabili analiticamente e quindi devono essere ottenute utilizzando algoritmi di tipo *Markov Chain Monte Carlo* (*MCMC*). A tale scopo sono stati proposti in letteratura diversi algoritmi:

- *Collapsed Gibbs sampler*: basato sullo schema delle urne di Polya, marginalizza rispetto a G (MacEachern, 1994).

- *Slice sampler*: partendo dalla rappresentazione *stick-breaking* riesce a ricondursi ad una forma finita tramite un duplice ricorso a variabili latenti (Walker, 2007).

- *Blocked Gibbs sampler*: basato sull'approssimazione di G tramite troncamento della rappresentazione *stick-breaking* (Ishwaran e James, 2001; Muliere e Tardella, 1998).

- *Retrospective sampler*: a differenza del precedente, permette di aggiungere in maniera adattiva, ma non eliminare, nuove componenti all'algoritmo a mano a mano che si rende necessario, senza il bisogno di ricorrere ad un troncamento (Papaspiliopoulos e Roberts, 2008).

Nel seguito viene presentato nel dettaglio l'algoritmo *Blocked Gibbs sampler* basato sui processi gaussiani. Questo algoritmo si basa appunto sull'approssimazione di G tramite troncamento della rappresentazione *stick-breaking*. Tale approccio trova giustificazione nel fatto che come mostrato i *random weights* w_h assegnati agli atomi funzionali decrescono geometricamente all'aumentare delle componenti della mistura e dunque risulta ragionevole sostituire la somma infinita di componenti nella rappresentazione *stick-breaking* con una somma sui primi N termini, ponendo $v_N = 1$, con N valore fissato sufficientemente grande. Si osservi che nel caso in cui la distribuzione di base del processo di Dirichlet sia condizionatamente coniugata, come accade quando si lavora con i processi gaussiani, le distribuzioni *full conditional* necessarie per la costruzione dell'algoritmo *Gibbs sampler* assumono delle semplici forme coniugate.

Nel dettaglio l'algoritmo si articola nei seguenti sei passi:

1. **Allocazione di ogni osservazione ad un gruppo**

   Sia S_i la variabile casuale multinomiale che indica il gruppo di appartenenza per l'i-esima curva. Essa assume un valore nell'insieme $\{1, \ldots, N\}$

con vettore di probabilità dato da $\pi_i = (\pi_{i1}, \ldots, \pi_{iN})$. Si ha che

$$\pi_{ih} = P\Big(S_i = h| - \Big) = \frac{w_h \prod_{t=1}^{T} \phi(y_i(t);\, \Theta_h(t), \sigma^2)}{\sum_{r=1}^{N} \Big(w_r \prod_{t=1}^{T} \phi(y_i(t);\, \Theta_r(t), \sigma^2)\Big)}$$

per $h = 1, \ldots, N$, dove $w_h = v_h \prod_{r<h}(1 - v_r)$ sono i *random weights*, mentre $\phi(a; b, c)$ indica la distribuzione di probabilità di una gaussiana con media b e varianza c, valutata in a.

2. **Aggiornamento dei pesi della rappresentazione *stick-breaking***

   Poiché la distribuzione a priori dei v_h è $Beta(1, \alpha)$, sfruttando la proprietà di coniugazione, si ottiene la seguente *full conditional*:

$$(v_h|-) \sim Beta(1 + n_h,\ \alpha + \sum_{r>h} n_r)$$

   per $h = \{1, \ldots, N-1\}$, dove n_h rappresenta il numero di osservazioni allocate nel gruppo h-esimo, $n_h = \sum_{i=1}^{n} I(S_i = h)$, con $I(\cdot)$ funzione indicatrice. Viene invece fissato $v_N = 1$ in modo da assegnare tutta la massa di probabilità restante all'ultimo gruppo.

3. **Aggiornamento degli atomi funzionali**

   La *full conditional* dei Θ_h è ancora normale multivariata con parametri aggiornati:

$$(\Theta_h|-) \sim N(\tilde{\mu}_h, \tilde{\Sigma}_h)$$
$$\tilde{\mu}_h = \mu + \mathcal{K}(\kappa_1, \kappa_2)\Big(\mathcal{K}(\kappa_1, \kappa_2) + \frac{\sigma^2}{n_h} I_T\Big)^{-1}(\bar{y}_h - \mu)$$
$$\tilde{\Sigma}_h = \mathcal{K}(\kappa_1, \kappa_2) - \mathcal{K}(\kappa_1, \kappa_2)\Big(\mathcal{K}(\kappa_1, \kappa_2) + \frac{\sigma^2}{n_h} I_T\Big)^{-1}\mathcal{K}(\kappa_1, \kappa_2)$$

   dove μ rappresenta la media a priori dei Θ_h, $\bar{y}_n$ il vettore media delle osservazioni allocate nel gruppo h-esimo e n_h come prima il numero di osservazioni che vi sono state allocate. Si osservi che nell'aggiornamento dell'h-esimo atomo funzionale contribuiscono solo le osservazioni che

sono state allocate a tale gruppo. Per ragioni computazionali di stabilità ed efficienza, a questo passaggio al posto di effettuare direttamente l'inversione della matrice è consigliabile passare per la decomposizione di Cholesky e sfruttare gli algoritmi per la risoluzione dei sistemi triangolari (Rasmussen e Williams, 2006).

4. **Aggiornamento dei parametri della funzione di covarianza**

   I parametri $\kappa = (\kappa_1, \kappa_2)$ che definiscono la funzione di covarianza non hanno una distribuzione *full conditional* riconducibile ad una forma nota dunque per il loro aggiornamento si ricorre ad un passo di *Metropolis* con passeggiata casuale uniforme su ogni componente del parametro in modo indipendente. La *full conditional* per il vettore $\boldsymbol{\kappa} = (\kappa_1, \kappa_2)^\top$ è proporzionale a

$$p(\boldsymbol{\kappa}|-) \propto p(\boldsymbol{\kappa})|\mathcal{K}(\boldsymbol{\kappa})|^{\frac{N}{2}} \exp\Big\{ -\frac{1}{2}\sum_{h=1}^{N}(\Theta_h - \tilde{\mu}_h)^\top \tilde{\Sigma}_h^{-1}(\Theta_h - \tilde{\mu}_h)\Big\}$$

   dove $p(\boldsymbol{\kappa})$ indica la distribuzione a priori congiunta per κ_1 e κ_2 mentre $\tilde{\mu}_h$ e $\tilde{\Sigma}_h$ sono le stesse quantità definite per l'aggiornamento dei Θ_h. Per ragioni computazionali di stabilità, il calcolo del determinante a questo passaggio può essere sostituito sfruttando ancora una volta la decomposizione di Cholesky, in particolare si ha che $|\mathcal{K}(\boldsymbol{\kappa})| = \prod_{i=1}^{n} L_{ii}^2$, dove $\mathcal{K}(\boldsymbol{\kappa}) = LL^\top$.

5. **Aggiornamento del parametro di varianza complessiva**

   Sfruttando le proprietà di coniugazione per il parametro σ^2 si ottiene la seguente *full conditional*

$$(\sigma^2|-) \sim IGa\Big(a_\sigma + \frac{nT}{2},\ b_\sigma + \frac{1}{2}\sum_{i=1}^{n}\sum_{t=1}^{T}\big(y_i(t) - \eta_i(t)\big)^2\Big)$$

   dove η_i è pari alla traiettoria Θ_h del gruppo a cui l'i-esima osservazione è stata assegnata, $\eta_i = \Theta_{S_i}$.

6. **Aggiornamento del parametro di concentrazione**

Poiché i *random weights* w_h hanno congiuntamente distribuzione di Dirichlet generalizzata, è possibile anche in questo caso sfruttare le proprietà di coniugazione (Escobar e West, 1995) e ottenere

$$(\alpha|-) \sim Ga\Big(a_\alpha + N - 1,\ b_\alpha - \sum_{h=1}^{N-1} \log(1 - v_h)\Big).$$

A partire da un'inizializzazione casuale dei parametri, questi sei passaggi vengono iterati R volte per ottenere, dopo un periodo di riscaldamento noto come *burn-in*, una serie di valori per ciascun parametro che permetta di studiare empiricamente le caratteristiche della distribuzione a posteriori. Si ricorda che i valori simulati da algoritmi di tipo *MCMC* sono dipendenti e presentano una densità marginale che converge alla distribuzione a posteriori che si vorrebbe analizzare. Risulta importante per questo controllare che tutte le catene siano arrivate a convergenza, effettuando eventualmente un filtraggio dei valori generati qualora presentino funzioni di autocorrelazione con valori elevati anche a ritardi alti della serie. Per un'introduzione agli algoritmi di tipo *MCMC* si veda, ad esempio, Robert e Casella (2010).

Capitolo 3

Modello basato sul processo *Probit Stick-Breaking*

3.1 Specificazione del modello

Già alla fine del Paragrafo 2.2 sul processo di Dirichlet si è detto che la rappresentazione *stick-breaking* del *DP* permette di generare nuove misure casuali sostituendo la caratteristica distribuzione $Beta(1, \alpha)$ con altre distribuzioni casuali. Sebbene la distribuzione Beta conferisca al processo di Dirichlet una serie di proprietà interessanti, in questo capitolo si vedrà come altre distribuzioni sui *random weights* portino ad avere distribuzioni a posteriori sui pesi della mistura molto più flessibili. L'idea alla base è quella di far dipendere la costruzione dei pesi da una serie di covariate specificando un opportuno modello di regressione. Tale approccio è applicabile tanto al processo di Dirichlet quanto al processo di Dirichlet funzionale in quanto va a modificare unicamente la struttura dei *random weights* w_h e non degli atomi della distribuzione di base G_0.

In Rodriguez e Dunson (2011) viene presentato nello specifico il processo *Probit Stick-Breaking* (*PSBP*) il quale sostituisce la distribuzione Beta con un modello *probit* anche se altre alternative sono percorribili, si veda ad esempio il processo *Logistic Stick-Breaking* (Ren et al., 2011). Nel modello basato sul

PSBP si assume che

$$G = \sum_{h=1}^{\infty} w_h \delta_{\Theta_h}, \quad \text{dove} \quad \Theta_h \sim G_0, \quad w_h = v_h \prod_{r<h} (1 - v_r)$$
$$\text{e} \quad v_h = \Phi(\alpha_h), \quad \text{con} \quad \alpha_h \sim N(\nu, \tau^2)$$

dove l'unica differenza rispetto a quanto specificato nel capitolo precedente consiste nella definizione dei v_h. Si osservi che ponendo $\nu = 0$ e $\tau^2 = 1$ si avrebbe $v_h \sim U(0, 1)$ che equivale ad un classico processo di Dirichlet con parametro di concentrazione $\alpha = 1$. In generale si ha che ν grandi positivi corrispondono ad α prossimi a 0 mentre ν grandi negativi ad α grandi. Per $\nu > 0$ si ottengono infatti distribuzioni dei v_h asimmetriche a sinistra che conferiscono grande importanza ai primi atomi del *PSBP* mentre per $\nu < 0$ si ottengono distribuzioni dei v_h asimmetriche a destra che portano a processi con concentrazione più elevata attorno alla distribuzione di base G_0.

Come nel processo di Dirichlet, anche nel processo *Probit Stick-Breaking* i pesi della mistura w_h decrescono geometricamente e dunque, oltre a rimanere contenuto il numero di gruppi identificati dal modello, risulta ragionevole mantenere l'approssimazione del processo basata sulle prime N componenti. Si può dimostrare che la distribuzione a posteriori del *PSBP* N-finito dimensionale converge in distribuzione a quella del processo non troncato basato su un numero infinito di componenti (Rodriguez e Dunson, 2011). Questo risultato è di particolare rilievo in quanto garantisce che i campioni ottenuti dalla distribuzione a posteriori del processo troncato possano essere utilizzati per generare inferenze arbitrariamente accurate su funzionali misurabili del processo infinito e garantisce inoltre la correttezza nell'utilizzo di algoritmi di tipo *Blocked Gibbs sampler*.

A differenza del processo di Dirichlet, il processo *Probit Stick-Breaking* non forma una famiglia coniugata, nel senso che la distribuzione a posteriori per G condizionata ai dati osservati non è a sua volta un processo *Probit Stick-Breaking* (Rodriguez e Dunson, 2011). Questo tuttavia non è un ostacolo per l'aggiornamento dei parametri i quali, nella maggior parte dei casi, presentano delle *full conditional* che sono riconducibili a distribuzioni note. Questo

consente di ottenere delle stime efficienti basate sul *Gibbs sampling* in algoritmi di tipo *MCMC*.

Se in assenza di variabili esplicative e per specifiche scelte di ν e τ^2 il *PSBP* si riduce ad un *DP* con parametro di concentrazione fissato, il caso interessante si incontra quando si hanno a disposizione una serie di variabili esplicative relative alle osservazioni. Con questa formulazione del processo infatti è possibile specificare un modello di regressione su α_h e far dipendere i pesi della rappresentazione *stick-breaking* dalle variabili esplicative. Di conseguenza, osservazioni con esplicative simili avranno *random weights* simili e saranno associate, a priori, con maggior probabilità agli stessi atomi.

Rimane da specificare la formulazione del modello di regressione sugli α_h, ad esempio assumendo $\alpha_h \sim N_n(X\beta_h, \Sigma_h)$, con X matrice di disegno di dimensione $n \times p$, β_h vettore dei p coefficienti di regressione parziali e Σ_h matrice di covarianza di dimensione $n \times n$. Sia il vettore β_h che la matrice Σ_h possono dipendere dal gruppo a cui α_h è riferito. Si noti che in questo caso i v_h, e dunque i *random weights* w_h, sono vettori di lunghezza n in quanto i pesi della rappresentazione *stick-breaking* cambiano da osservazione a osservazione in base alle rispettive variabili esplicative.

3.2 Modellazione della dipendenza spaziale e temporale

Riprendendo il contesto applicativo in ambito demografico di questo lavoro, se l'interesse è modellare l'effetto del tempo e dello spazio nell'evoluzione delle curve di mortalità, è necessario stabilire come inserire le variabili esplicative relative all'anno di riferimento, alla latitudine e alla longitudine nella specificazione del modello *probit*.

Per modellare la variazione in media e dunque l'effetto di grande scala è possibile inserire le variabili esplicative nel predittore lineare $X\beta_h$, procedendo ad eventuali estensioni polinomiali di queste per cogliere anche *trend* temporali e/o spaziali di tipo non lineare. Se si imposta un semplice modello di regressione lineare con errori omoschedastici e indipendenti la stima della

matrice di covarianza si riduce alla stima di un solo parametro, posto $\Sigma_h = \tau_h^2 I_n$. Un'alternativa che può risultare utile per la modellazione dell'effetto di grande scala della latitudine e della longitudine è quella di utilizzare delle *spline* prodotto tensoriale (Azzalini e Scarpa, 2009). In questo modo si costruisce una base di funzioni data dal prodotto di basi di funzioni unidimensionali relative alle due variabili prese singolarmente. Così facendo si riesce a cogliere la presenza di eventuali effetti di interazione fra le due direzioni dello spazio oltre che andamenti di tipo non lineare. La scelta delle *spline* prodotto tensoriale presenta il vantaggio di essere altamente flessibile ma il numero di coefficienti β_h da stimare diventa presto elevato aumentando il numero di nodi e il grado delle singole *splines*.

Una scelta differente è quella di assumere che la componente residuale che descrive la variazione di piccola scala sia di tipo stocastico. Invece di definire per α_h un modello lineare con errori omoschedastici e indipendenti, è possibile specificare una matrice di covarianza Σ_h piena, assumendo una qualche struttura. Si può utilizzare ad esempio la medesima struttura esponenziale assunta per la funzione di covarianza del processo gaussiano del *FDP*. Pensando a titolo esemplificativo alla trattazione della dipendenza spaziale fra le curve, scegliere una funzione di covarianza esponenziale consente di ottenere una superficie liscia nello spazio, ipotizzando che la dipendenza fra le curve osservate diminuisca all'aumentare della loro distanza spaziale. Tale distanza può essere espressa ad esempio in termini di distanza euclidea fra le coordinate medie di latitudine e longitudine delle rispettive aree geografiche o in termini dicotomici al fine di discriminare i casi di curve collocate in aree geografiche confinanti o meno. Qualunque altro criterio sensato basato su caratteristiche del territorio può essere utilizzato per la definizione della matrice delle distanze spaziali. Assumendo una funzione di covarianza esponenziale generalizzata (Dunson e Herring, 2006) con vettore dei parametri $\boldsymbol{\kappa}^* = (\kappa_1^*, \kappa_2^*)^\top$, si ha che

$$\Sigma_h = \Sigma_h(\boldsymbol{\kappa}^*) = \kappa_1^* \exp\{-\kappa_2^*(d - d')^2\} + \zeta\kappa_1^* I(d = d')$$

dove il termine $(d-d')^2$ indica la matrice di distanze spaziali. Per la trattazione

della dipendenza temporale basterà sostituire alla matrice delle distanze spaziali una matrice di distanze temporali, indicata con $(a-a')^2$. Quest'ultima risulta di immediata definizione in quanto una curva di mortalità è associata tipicamente ad uno specifico anno.

Si potrebbe essere interessati infine alla trattazione congiunta della variazione di piccola scala sia rispetto al tempo che allo spazio. In questo caso è possibile definire una matrice di distanze spazio-temporali data dal prodotto delle due matrici di distanze, la prima che dipende solo dalla distanza spaziale e la seconda che dipende solo dalla distanza temporale. Assumendo un parametro di magnitudine comune alle due matrici per evitare problemi di sovra parametrizzazione e non identificabilità dei parametri della matrice, è possibile definire

$$\Sigma_h(\boldsymbol{\kappa}') = \kappa_1' \exp\{-\kappa_2'(d-d')^2 - \kappa_3'(a-a')^2\} + \zeta\kappa_1' I(d=d')I(a=a')$$

con $\boldsymbol{\kappa}' = (\kappa_1', \kappa_2', \kappa_3')^\top$ vettore di parametri positivi dove κ_1' regola la magnitudine dell'intera matrice di distanze spazio-temporali mentre κ_2' e κ_3' quanto deve essere liscia la superficie rispetto alla distanza spaziale e temporale rispettivamente.

Per completare la specificazione del modello si definisce in aggiunta una distribuzione a priori per il vettore dei coefficienti del modello *probit* $\beta_h \sim N_p(\mu_\beta, \Sigma_\beta)$, con $\Sigma_\beta = \sigma_\beta^2 I_p$, e delle distribuzioni a priori Gamma per i parametri delle matrici Σ_h o, in alternativa, una distribuzione Gamma Inversa per τ_h^2. Una scelta di distribuzione a priori diffusa, quindi sufficientemente non informativa, per β_h può essere quella di fissare $\mu_\beta = 0$ e σ_β^2 grande. Per le distribuzioni Gamma e Gamma Inversa valgono le stesse considerazioni fatte per le distribuzioni a priori del modello basato sul *FDP*.

Si noti che in questo modello non si ha più il parametro di concentrazione α che si aveva nel processo di Dirichlet in quanto la distribuzione $Beta(1, \alpha)$ viene sostituita dal modello *probit* specificato. In questo modello infatti sono i parametri β_h a regolare la concentrazione delle estrazioni del processo attorno alla distribuzione di base. Posto x_i il vettore delle variabili esplicative dell'i-esima osservazione, valori piccoli di $x_i^\top \beta_h$ implicano estrazioni molto

concentrate attorno alla distribuzione di base in quanto gli α_{ih} saranno piccoli e di conseguenza i v_{ih} prossimi allo zero.

3.3 Algoritmo di stima

Nel seguito viene presentata una versione modificata dell'algoritmo *Blocked Gibbs sampler* visto nel Paragrafo 2.3.2 per il *FDP* che sostituisce al passaggio relativo all'aggiornamento del parametro di concentrazione α un passaggio per l'aggiornamento dei parametri del modello *probit*. Nel dettaglio l'algoritmo si articola nei seguenti passi:

1. **Allocazione di ogni osservazione ad un gruppo**

   Sia S_i la variabile casuale multinomiale che indica il gruppo di appartenenza per l'i-esima curva. Essa assume un valore nell'insieme $\{1, \ldots, N\}$ con vettore di probabilità dato da $\pi_i(x_i) = (\pi_{i1}(x_i), \ldots, \pi_{iN}(x_i))$. In questo caso si è indicato $\pi_i(x_i)$ per sottolineare la dipendenza tramite i $v_{ih}(x_i)$ del vettore delle esplicative x_i nella costruzione dei pesi. Si ha che

   $$\pi_{ih}(x_i) = P\Big(S_i = h| - \Big) = \frac{w_{ih}(x_i) \prod_{t=1}^{T} \phi(y_i(t);\, \Theta_h(t), \sigma^2)}{\sum_{r=1}^{N} \Big(w_{ir}(x_i) \prod_{t=1}^{T} \phi(y_i(t);\, \Theta_r(t), \sigma^2)\Big)}$$

   per $h = 1, \ldots, N$, dove $w_{ih}(x_i) = v_{ih}(x_i) \prod_{r<h}(1 - v_{ir}(x_i))$ sono i *random weights* che dipendono dal vettore delle esplicative x_i.

2. **Aggiornamento delle quantità per la costruzione dei *random weights***

   Per l'aggiornamento degli α_h alla base della costruzione dei *random weights*, Chung e Dunson (2009) utilizzano un approccio di *data augmentation* introducendo una variabile latente $z_{ih} \sim N(\alpha_{ih}, 1)$. A questo punto si pone $S_i = h$ se e solo se $z_{ih} > 0$ e $z_{ir} < 0$ per ogni $r < h$. In

questo modo si ottiene correttamente che

$$\begin{aligned} P(S_i = h) &= P(z_{ih} > 0, z_{ir} < 0 \text{ per } r < h) \\ &= \Phi(\alpha_{ih}) \prod_{r<h} (1 - \Phi(\alpha_{ir})) \\ &= w_{ih}. \end{aligned}$$

Condizionatamente al valore α_{ih} e alla variabile S_i che indica il gruppo di appartenenza, è possibile campionare il valore di z_{ih} dalla sua *full conditional*

$$(z_{ih}|-) \sim \begin{cases} N(\alpha_{ih}, 1)_{\mathbb{R}^-} & \text{se} \quad S_i > h \\ N(\alpha_{ih}, 1)_{\mathbb{R}^+} & \text{se} \quad S_i = h \end{cases}$$

dove viene indicata con $N(a, b)_\Omega$ una distribuzione normale di media a e varianza b troncata sull'insieme Ω. Il valore di α_{ih} viene quindi campionato condizionatamente al valore della variabile latente e sfruttando le proprietà di coniugazione della distribuzione normale a priori di $(\alpha_h|\beta_h, \Sigma_h) \sim N(X\beta_h, \Sigma_h)$.

3. **Aggiornamento dei parametri del modello *probit***

A questo passaggio è necessario fare una distinzione in base al modello *probit* scelto. Nel caso più semplice in cui $\Sigma_h = \tau_h^2 I_n$, questo diventa un modello lineare normale bayesiano per il quale sono note le *full conditional* dei parametri β_h e τ_h^2. In particolare si ha che

$$(\tau_h^2|-) \sim IGa\left(a_\tau + \frac{n'_h}{2},\ b_\tau + \frac{1}{2}\left(\alpha'_h - X'\beta_h\right)^\top \left(\alpha'_h - X'\beta_h\right)\right)$$

$$(\beta_h|-) \sim N_p\left(\left(X'^\top X' + \frac{1}{\tau_h^2} I_p\right)^{-1} \left(\frac{1}{\tau_h^2}\mu_\beta + X'^\top \alpha'_h\right), \tau_h^2 \left(X'^\top X' + \frac{1}{\tau_h^2} I_p\right)^{-1}\right)$$

dove α'_h è il vettore di lunghezza n'_h dei soli valori di α_h in corrispondenza delle osservazioni con $S_i >= h$, e X' la matrice di disegno con le righe corrispondenti. Contribuiscono alla stima dunque solo le osservazioni che non sono già state allocate. Si noti che scegliendo delle priori sufficientemente non informative, ossia con σ_β^2 grande, la posteriori di

$\beta_h|-$ che si ottiene è equivalente all'approssimazione quadratica dello stimatore di massima verosimiglianza. Con una scelta più informativa per gli iperparametri delle distribuzioni a priori si ottiene invece un β_h con valore atteso condizionato dato dalla media pesata fra la stima di massima verosimiglianza e il valore della media a priori μ_β. Viceversa nel caso venga specificato un modello *probit* con matrice Σ_h con struttura esponenziale non si riesce a ricondursi ad una *full conditional* nota e dunque si ricorre ad un passo di *Metropolis* con passeggiata casuale uniforme su ogni componente del parametro $(\beta_h, \boldsymbol{\kappa}_h)$. La *full conditional* è proporzionale a

$$p(\beta_h, \boldsymbol{\kappa}_h|-) \propto p(\beta_h, \boldsymbol{\kappa}_h)|\Sigma'_h(\boldsymbol{\kappa}_h)|^{\frac{1}{2}n'_h} \exp\Big\{-\frac{1}{2}(\alpha'_h - X'\beta_h)^\top \Sigma'_h(\boldsymbol{\kappa}_h)^{-1}(\alpha'_h - X'\beta_h)\Big\}$$

dove con $p(\beta_h, \boldsymbol{\kappa}_h)$ viene indicata la distribuzioni a priori congiunta dell'intero vettore di parametri che regola il modello *probit* mentre con $\Sigma'_h(\boldsymbol{\kappa}_h)$ la matrice di covarianza $n'_h \times n'_h$ con le righe e le colonne filtrate in accordo con α'_h e X'.

4. **Aggiornamento degli atomi funzionali**

   Questo passaggio risulta uguale a quello dell'algoritmo per la stima dei parametri del modello basato sul *FDP* ma per completezza lo si riporta ugualmente. La *full conditional* dei Θ_h è ancora normale multivariata con parametri aggiornati:

$$(\Theta_h|-) \sim N(\tilde{\mu}_h, \tilde{\Sigma}_h)$$
$$\tilde{\mu}_h = \mu + \mathcal{K}(\kappa_1, \kappa_2)\Big(\mathcal{K}(\kappa_1, \kappa_2) + \frac{\sigma^2}{n_h} I_T\Big)^{-1}(\bar{y}_h - \mu)$$
$$\tilde{\Sigma}_h = \mathcal{K}(\kappa_1, \kappa_2) - \mathcal{K}(\kappa_1, \kappa_2)\Big(\mathcal{K}(\kappa_1, \kappa_2) + \frac{\sigma^2}{n_h} I_T\Big)^{-1}\mathcal{K}(\kappa_1, \kappa_2)$$

   dove μ rappresenta la media a priori dei Θ_h, $\bar{y}_n$ il vettore media delle osservazioni allocate nel gruppo h-esimo e n_h come prima il numero di osservazioni che vi sono state allocate. Si osservi che nell'aggiornamento

dell'h-esimo atomo funzionale contribuiscono solo le osservazioni che sono state allocate a tale gruppo. Si osservi che come detto nell'algoritmo di stima del modello basato sul *FDP* da un punto di vista computazionale questo passaggio può risultare molto oneroso in quanto comporta l'inversione di matrici di dimensione $n \times n$ e dunque da un punto di vista di implementazione devono essere adottate alcune accortezza.

5. **Aggiornamento dei parametri della funzione di covarianza $\mathcal{K}$**

   Analogamente a quanto visto per il modello basato sul *FDP*, per l'aggiornamento dei parametri $\kappa = (\kappa_1, \kappa_2)$ si ricorre ad un passo di *Metropolis* con passeggiata casuale uniforme su ogni componente del parametro in modo indipendente. La *full conditional* per κ è proporzionale a

$$p(\kappa|-) \propto p(\kappa)|\mathcal{K}(\kappa_1, \kappa_2)|^{\frac{N}{2}} \exp\left\{ -\frac{1}{2} \sum_{h=1}^{N} (\Theta_h - \mu)^\top \mathcal{K}(\kappa_1, \kappa_2)^{-1} (\Theta_h - \mu) \right\}$$

   dove $p(\kappa)$ indica la distribuzione a priori per κ.

6. **Aggiornamento del parametro di varianza complessiva σ^2**

   Come per il modello basato sul *FDP*, anche in questo caso si sfruttano le proprietà di coniugazione del parametro σ^2 per ottiene la seguente *full conditional*

$$(\sigma^2|-) \sim IGa\Big(a_\sigma + \frac{nT}{2},\ b_\sigma + \frac{1}{2} \sum_{i=1}^{n} \sum_{t=1}^{T} \big(y_i(t) - \eta_i(t) \big)^2 \Big)$$

   dove η_i è pari alla traiettoria Θ_h del gruppo a cui l'i-esima osservazione è stata assegnata, $\eta_i = \Theta_{S_i}$.

Anche per questo modello partendo da un'inizializzazione casuale dei parametri e dopo il periodo di riscaldamento, i valori generati dalle catene potranno essere utilizzati per studiare empiricamente le caratteristiche della distribuzione a posteriori.

Capitolo 4

Applicazione ai dati

4.1 I dati

I dati analizzati in questo capitolo sono gratuitamente reperibili all'interno del *Human Mortality Database* (Barbieri et al., 2015), nel quale vengono rese disponibili le tavole di mortalità per 41 paesi del mondo a scopi di monitoraggio e ricerca scientifica. Per le ragioni indicate alla fine del Paragrafo 1.2.1, fra le quantità riportate nella tavola di mortalità si sceglie di modellare la funzione d_x, rappresentante la distribuzione dei decessi per età. Si specifica che in tutte le analisi seguenti per comodità, la distribuzione dei decessi per età considerata è quella totale, ovvero per maschi e femmine assieme, nonostante sia noto che questi presentino curve di mortalità fra loro differenti. Tutti i modelli che verranno discussi possono essere applicati, con le dovute variazioni sui parametri di regolazione dell'algoritmo, alle distribuzioni dei decessi per età per la popolazione dei maschi e delle femmine in modo separato. In fase di stima, inoltre, la funzione d_x viene divisa per un fattore di 10^5 in modo da ottenere una curva che integri ad uno.

L'obiettivo dell'analisi è quello di valutare l'evoluzione temporale e spaziale della mortalità mediante l'applicazione dei modelli bayesiani non parametrici presentati nei Capitoli 2 e 3. Per lo studio dell'evoluzione temporale ci si concentra sullo stato italiano e si valuta come è cambiata la distribuzione delle morti per età negli anni dal 1872 al 2018 (Figura 4.1), intero arco temporale

per il quale sono disponibili i dati relativi all'Italia. Graficamente si osserva una progressiva riduzione della mortalità infantile, dovuta al miglioramento delle condizioni igenico-sanitarie, ed un avanzamento dell'età modale che si è lentamente assestata attorno agli 88 anni. Anche la compressione della curva attorno a quest'ultima moda è aumentata con il passare degli anni, nei quali la componente di mortalità accidentale e prematura è andata via via riducendosi. Per lo studio dell'evoluzione spaziale, invece, si utilizzano i dati dei 30 stati europei presenti all'interno del *Human Mortality Database* relativi all'anno 2013, ultimo anno per il quale sono disponibili i dati di tutti e 30 gli stati. La distribuzione delle morti per età di questi paesi è rappresentata in Figura 4.2. Anche se l'elevata numerosità delle curve su scala cromatica non aiuta la rappresentazione grafica, si può osservare una certa differenza in termini di forma della curva fra i paesi più e meno sviluppati. Gli stati con un livello di benessere inferiore, scarse condizioni igenico-sanitarie e maggior povertà presentano un anticipo sul valore modale dell'età adulta ed una mortalità prematura più elevata rispetto agli altri stati.

4.2 Valutazione della convergenza

Per tutti i modelli sono state eseguite 10 000 iterazioni dell'algoritmo, dove le prime 2 500 vengono considerate di riscaldamento e dunque scartate. La rappresentazione *stick-breaking* viene troncata a $N = 10$ componenti. Il valore scelto, che determina il numero massimo di gruppi ottenibili, risulta adeguato in quanto il modello identifica al più cinque gruppi in tutti gli scenari. La convergenza viene monitorata tramite un'analisi visiva dei grafici delle traiettorie, riportati in Appendice A, e due diagnostiche: l'*effective sample size* e il *potential scale reduction factor* (Robert e Casella, 2010). L'*effective sample size* fornisce un'indicazione della perdita di informazione dovuta all'utilizzo di un campione di valori dipendenti rispetto ad uno di valori indipendenti. Il valore che restituisce infatti è pari alla numerosità stimata di un campione di valori indipendenti con stesso contenuto informativo del campione *MCMC* generato. Sono desiderabili valori alti di questa diagnostica

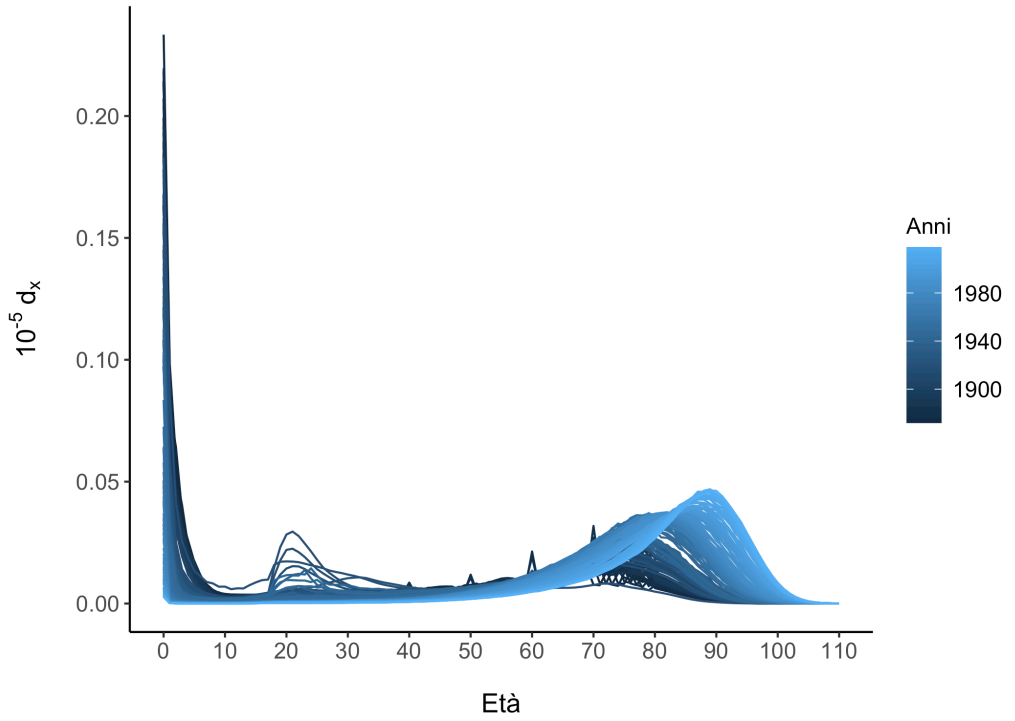


Figura 4.1: Distribuzione delle morti per età della popolazione italiana dal 1872 al 2018.

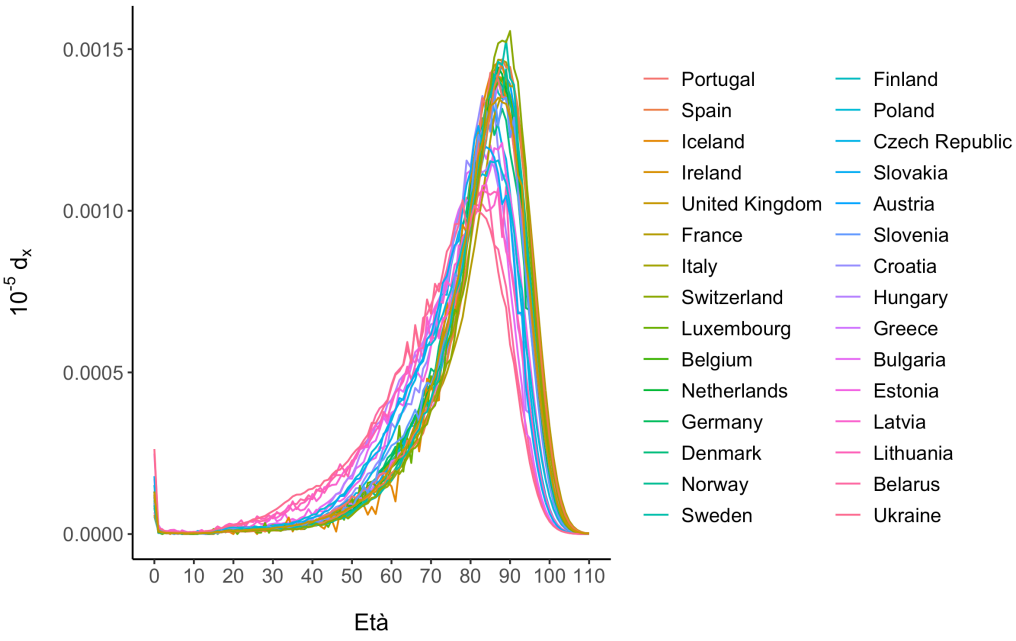


Figura 4.2: Distribuzione delle morti per età dell'anno 2013 di 30 stati europei.

in relazione alla lunghezza della catena. Valori bassi sono indice della presenza di forte autocorrelazione nella serie di valori ottenuti, che può essere migliorata modificando i parametri dell'algoritmo o eseguendo un filtraggio dei valori generati. Il *potential scale reduction factor*, noto anche come diagnostica di Gelman e Rubin, necessita del calcolo per ogni parametro di $M \geq 2$ catene di lunghezza $2R$ costruite a partire da diversi valori iniziali. Scartati i primi R valori di ciascuna serie, confronta la varianza entro e tra le catene di ciascun parametro. Valori maggiori di 1.2 indicano una cattiva convergenza alla distribuzione stazionaria.

Quando si utilizzano modelli bayesiani non parametrici e si è interessati al raggruppamento che il modello induce o all'interpretazione di parametri che dipendono dal gruppo di riferimento, come ad esempio avviene nel modello basato sul *PSBP*, un aspetto al quale bisogna prestare attenzione è quello che va sotto il nome di *label switching*, ovvero "scambio di etichette". Ad ogni iterazione infatti si ottiene un raggruppamento che può differire dai raggruppamenti precedenti per numero e per significato dei gruppi identificati. In altre parole, il gruppo etichettato con "1" alla r-esima iterazione potrebbe essere etichettato con "2" all'iterazione successiva e viceversa per l'altro gruppo, questo perché i corrispettivi modelli mistura che ne stanno alla base presentano la stessa verosimiglianza. Prima del calcolo di altre diagnostiche sulle catene risulta importante quindi l'analisi delle traiettorie dei parametri, le quali possono presentare uno o più punti di salto. Se non trattato, il *label switching* può condurre ad un'errata valutazione della convergenza alla distribuzione finale. Per una trattazione più dettagliata del problema e di una sua possibile soluzione si veda, ad esempio, Stephens (2000).

4.3 La mortalità nel tempo in Italia

4.3.1 Applicazione modello basato sul *FDP*

I risultati ottenuti dall'applicazione del modello basato sul processo di Dirichlet funzionale sono rappresentati in Figura 4.3. Nello specifico, la Figura 4.3(a) rappresenta gli atomi funzionali dei cinque gruppi identificati

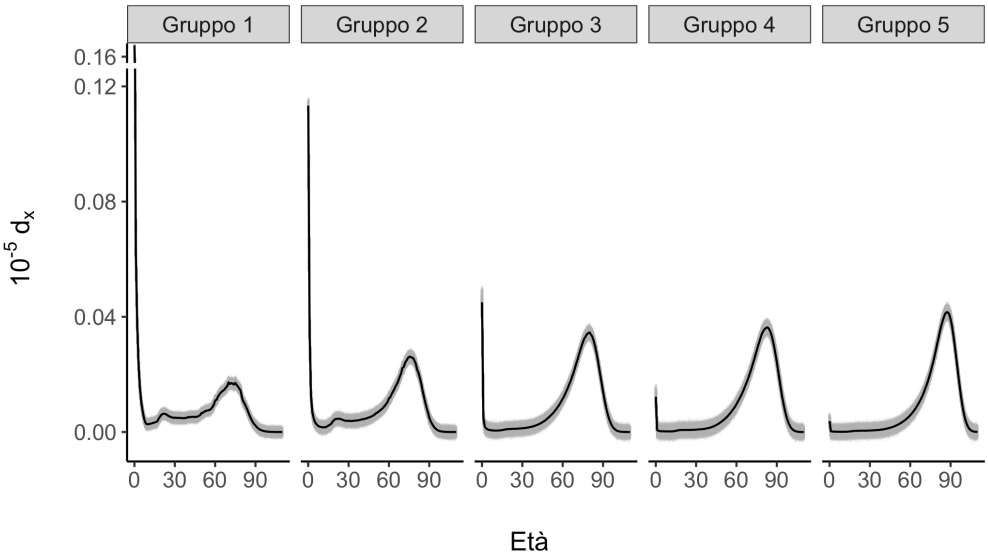
dal modello. Si osservi che grazie all'utilizzo di una funzione di covarianza esponenziale per il processo gaussiano che regola l'estrazione di questi atomi, essi risultano lisci e possono essere utilizzati per lisciare le osservazioni allocate in quel gruppo senza la necessità di ricorrere ad un lisciamento preliminare dei dati. In Figura 4.3(b) viene mostrato il raggruppamento ottenuto mentre in Figura 4.3(c) la distribuzione degli anni entro ciascun gruppo. Come si può notare vi è un'evoluzione della mortalità quasi lineare nel tempo. I cinque gruppi appaiono ben separati, in Tabella 4.1 si riporta la loro esatta composizione, specificando per ciascun gruppo la sequenza di anni che il modello vi alloca.

Si osserva che il quinquennio 1915-1919 viene allocato nel primo gruppo nonostante potrebbe appartenere temporalmente al secondo, che inizia già nel corso degli anni precedenti. Questo ritorno al gruppo delle curve di mortalità più antiche è un chiaro segno della retrocessione in termini di mortalità che vi è stata con l'ingresso dell'Italia nella Prima Guerra Mondiale.

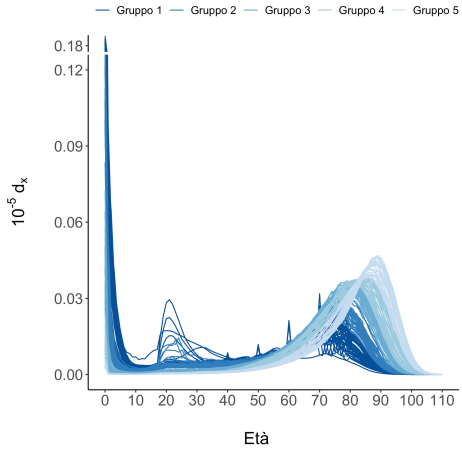
Osservando i grafici delle traiettorie in Figura A.1 non sembrano esserci evidenze contro la convergenza di nessuna catena. Anche le diagnostiche riportate in Tabella 4.2 non evidenziano problemi particolari. Per il loro calcolo sono state costruite due catene di 10 000 iterazioni ciascuna, di cui 5 000 di riscaldamento, facendo partire l'algoritmo di stima da due inizializzazioni differenti. La numerosità complessiva del campione *MCMC* generato con cui confrontare l'*effective sample size* è di $5\,000 + 5\,000 = 10\,000$.

Tabella 4.1: Mortalità in Italia - Composizione dei gruppi del modello basato sul *FDP*.

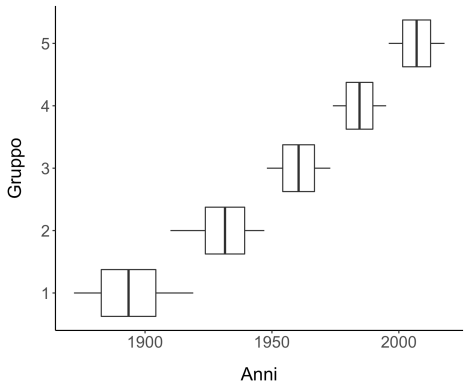
	Anni
Gruppo 1	1872 - 1909, 1911, 1915 - 1919
Gruppo 2	1910, 1912 - 1914, 1920 - 1947
Gruppo 3	1948 - 1986
Gruppo 4	1987 - 2018



(a) Atomi funzionali estratti



(b) Raggruppamento ottenuto



(c) Suddivisione anni

Figura 4.3: Mortalità in Italia - Risultati del modello basato sul *FDP*.

Tabella 4.2: Mortalità in Italia - Controllo convergenza modello basato sul *FDP* per il parametro di varianza complessiva σ^2, il parametro di concentrazione α e il parametro κ che regola la funzione di covarianza esponenziale in cui si è fissato il parametro relativo alla magnitudine.

Diagnostica	σ^2	α	κ
Effective sample size (su 10 000)	9 267	3 779	1 993
Potential scale reduction factor	1	1	1

4.3.2 Applicazione modello basato sul *FPSBP*

Alla luce dei risultati ottenuti con il modello basato sul *FDP* si intende stimare un modello basato sul processo *Probit Stick-Breaking* funzionale (*FPSBP*) per capire se un modello di regressione che sfrutta l'informazione temporale riesce a spiegare l'allocazione delle osservazioni nei diversi gruppi. Per i risultati emersi in precedenza si ipotizza un modello *probit* con un semplice *trend* lineare nel tempo.

La Figura 4.4 mostra i risultati del modello basato sul *FPSBP*. Si osserva che gli atomi funzionali estratti sono perfettamente in linea con quelli identificati dal modello precedente e anche il raggruppamento delle osservazioni che ne si deriva è lo stesso (Tabella 4.3). I grafici con le traiettorie dei parametri riportati in Appendice in Figura A.2 e in Figura A.3 e le diagnostiche di convergenza calcolate non mostrano particolari evidenze contrarie al raggiungimento della distribuzione limite. In Tabella 4.4 per i parametri che regolano il modello *probit* viene riportato il valore medio che la diagnostica assume negli N gruppi.

Rispetto al modello basato sul *FDP*, questo modello è in grado di sfruttare le informazioni aggiuntive di cui si dispone sulle curve di mortalità, come ad esempio l'anno a cui ognuna è riferita, e inserirle nel modello mediante la distribuzione a priori sui *random weights*. La distribuzione dei *random weights* con parametri aggiornati è mostrata in Figura 4.5, dove la curva rappresentata viene ottenuta al variare di $i = 1, \ldots, n$ e ordinando temporalmente le osservazioni sull'asse delle ascisse per ciascuno dei cinque gruppi identificati.

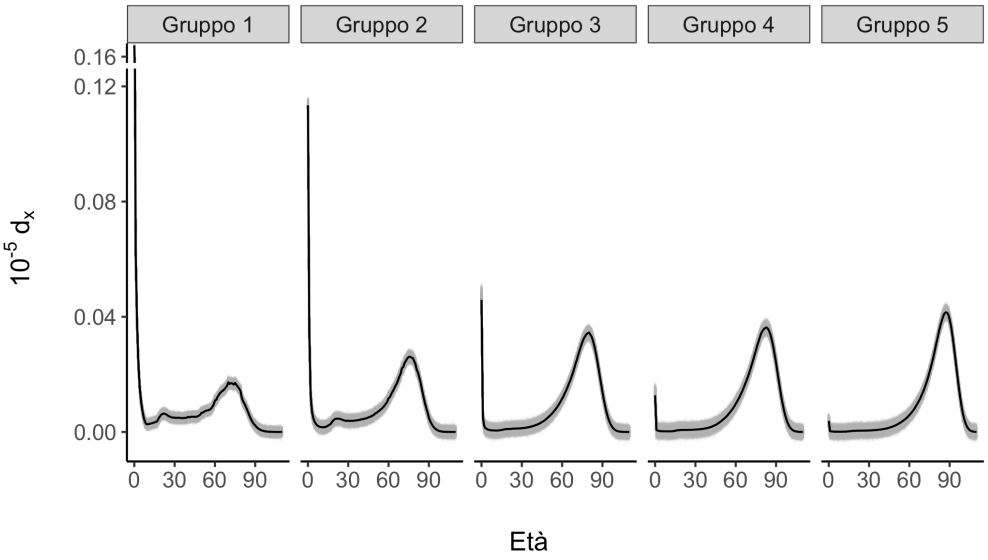
Tabella 4.3: Mortalità in Italia - Composizione dei gruppi del modello basato sul *FPSBP*.

	Anni
Gruppo 1	1872 - 1909, 1911, 1915 - 1919
Gruppo 2	1910, 1912 - 1914, 1920 - 1947
Gruppo 3	1948 - 1986
Gruppo 4	1987 - 2018

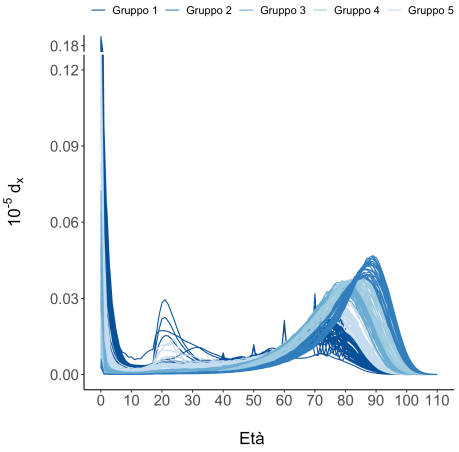
Tabella 4.4: Mortalità in Italia - Controllo convergenza modello basato sul *FPSBP* per il parametro di varianza complessiva σ^2, il parametro κ che regola la funzione di covarianza esponenziale in cui si è fissato il parametro relativo alla magnitudine e i parametri β_{0h}, β_{1h} e τ_h^2 del modello *probit* dei quali si riporta un valore medio della diagnostica.

Diagnostica	σ^2	κ	β_0	β_1	τ^2
Effective sample size (su 10 000)	9 555	2 330	1 501	1 701	2 234
Potential scale reduction factor	1	1	1.08	1.12	1.02

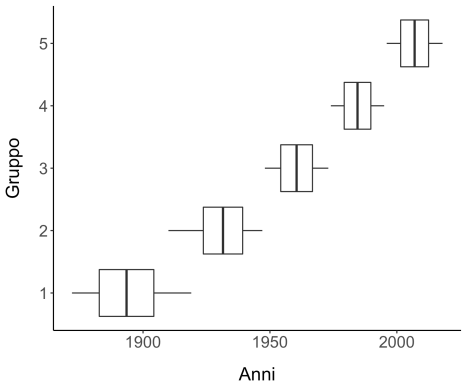
Un modello di questo tipo favorisce a priori il raggruppamento della nuova curva verso curve con valori delle variabili esplicative simili, e quindi in questo caso vicine temporalmente, ancora prima di osservarne la forma funzionale. Chiaramente poi questa informazione aggiuntiva veicolata tramite la distribuzione a priori viene integrata con l'informazione proveniente dai dati una volta misurata e osservata la forma funzionale della curva. Riprendendo ad esempio il quinquennio 1915-1919 in cui c'è stata la Prima Guerra Mondiale si può vedere come queste curve vengano allocate all'interno del primo gruppo nonostante la distribuzione dei *random weights*, in particolare per gli ultimi tre anni di questi, risulti più a favore del secondo gruppo. Questo significa che nonostante si trovino temporalmente più vicine alle curve di mortalità del secondo gruppo, la loro forma funzionale e le loro caratteristiche in termini di mortalità sono risultate molto più simili a quelle del primo gruppo. Il fatto che in presenza di una forte indicazione da parte dei dati i due modelli arrivino agli stessi risultati pare del tutto ragionevole, differendo unicamente per la specificazione della distribuzione a priori dei *random weights*.



(a) Atomi funzionali estratti



(b) Raggruppamento ottenuto



(c) Suddivisione anni

Figura 4.4: Mortalità in Italia - Risultati del modello basato sul *FPSBP*.

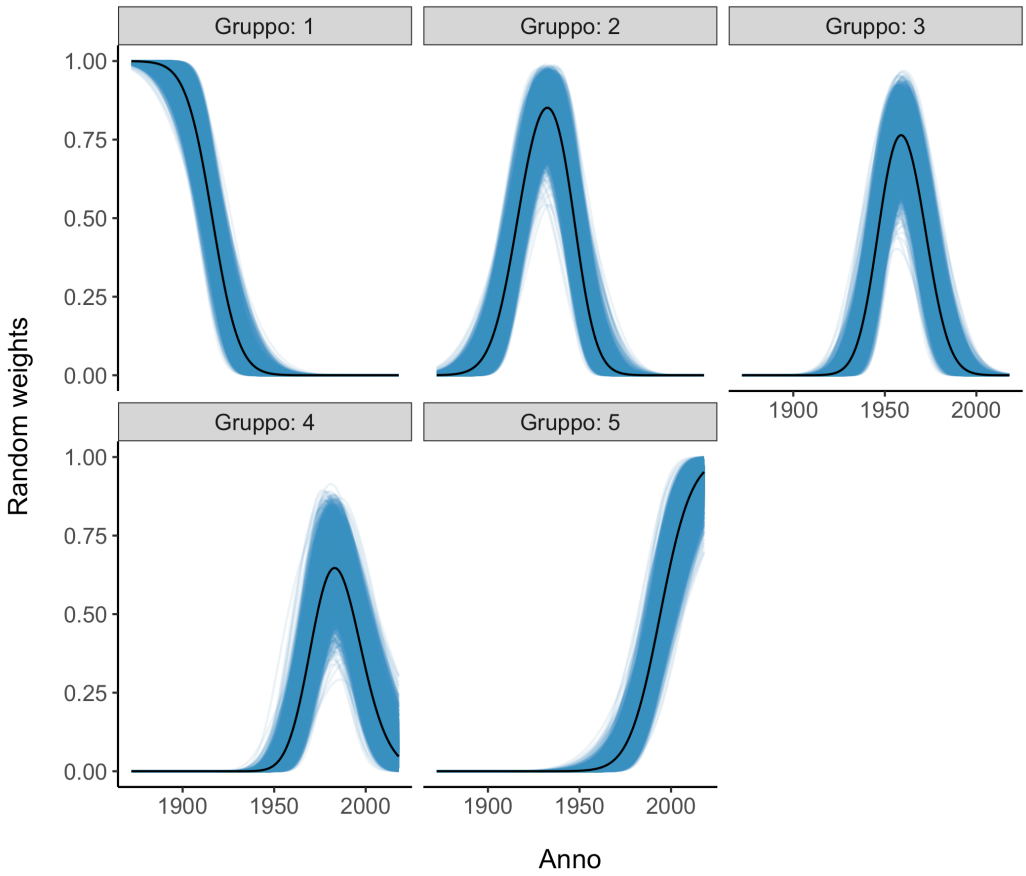


Figura 4.5: *Random weights* del *FPSBP* associati alla curva di ciascun anno per i cinque gruppi identificati.

4.4 Robustezza della distribuzione a priori degli atomi funzionali

Sia il modello basato sul *FDP* e sia quello basato sul processo *FPSBP*, assumono che la distribuzione di base sia un processo gaussiano con funzione media μ e funzione di covarianza $\mathcal{K}$, ovvero che la distribuzione a priori per gli atomi funzionali sia $\Theta_h \sim GP(\mu, \mathcal{K})$. Una distribuzione a priori diffusa con una funzione media costante e una funzione di covarianza che induce estrazioni con una variabilità elevata potrebbe sembrare sufficientemente non informativa e dunque una scelta adeguata. Si osservi tuttavia che per come sono costruiti gli algoritmi *Blocked Gibbs sampler*, quando il valore n_h pari al numero di osservazioni allocate nell'h-esimo gruppo è nullo, ovvero quando si è in presenza di un gruppo in cui non vi sono allocate osservazioni, la *full conditional* dei Θ_h è pari alla loro distribuzione a priori. Questo significa che gli atomi funzionali dei gruppi vuoti vengono generati dalla distribuzione a priori che se risulta troppo distante dai dati, produrrà estrazioni inverosimili che andranno a penalizzare le probabilità π_{ih} di quel gruppo. Ne consegue l'incapacità dell'algoritmo di allocare nuovamente osservazioni sui gruppi che si svuotano. Questo comporta che l'eliminazione di un gruppo nelle prime iterazioni sia di fatto irreversibile, andando a modificare la convergenza alla distribuzione finale.

Utilizzando a solo scopo illustrativo i dati della mortalità in Italia, viene condotto un semplice studio in cui la stima del modello basato sul *FPSBP* viene replicata per 20 volte per diverse inizializzazioni dell'algoritmo. In Figura 4.6 si mostra come il numero di gruppi identificati dal modello cambi se si utilizza per μ una distribuzione uniforme, che corrisponde ad assumere che la probabilità di morte sia la stessa per tutte le età. Oltre a sembrare poco ragionevole, si osserva come il numero di gruppi identificati dal modello nel corso delle iterazioni non cresca mai, finendo per trovare sempre uno o due gruppi. Utilizzando invece per μ una forma funzionale semplice ma sensata, come quella fornita da un modello di Siler con parametri fissati, si ottiene una convergenza verso l'identificazione dei cinque gruppi mostrati nel paragrafo

precedente. Con più iterazioni ci si aspetta che anche le 3 simulazioni che hanno identificato tre soli gruppi convergano a cinque.

Una scelta più informativa e più sensata della distribuzione a priori risulta in questo senso necessaria per consentire l'allocazione delle curve in nuovi gruppi. Questo inoltre sembra migliorare anche la convergenza nell'identificazione del numero di gruppi presenti in un insieme di dati, quantità che il processo di Dirichlet non riesce a stimare in modo consistente (Miller e Harrison, 2018).

4.5 La mortalità nello spazio in Europa

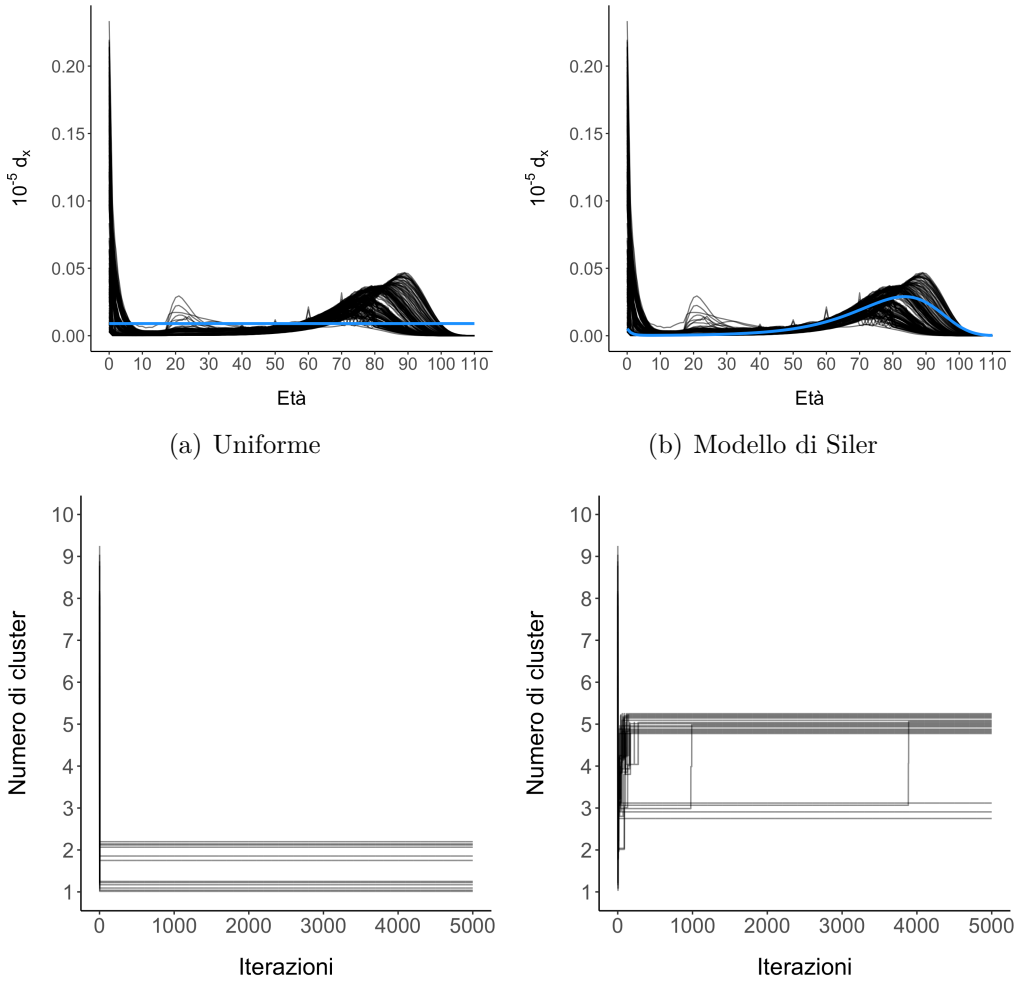
4.5.1 Applicazione modello basato sul *FDP*

I risultati ottenuti dall'applicazione del modello basato sul processo di Dirichlet funzionale sono rappresentati in Figura 4.7. I due gruppi che il modello identifica appaiono ben separati e con forme funzionali differenti. Il primo gruppo è quello che si può riferire ai paesi più sviluppati con curve di mortalità altamente compresse attorno al valore modale che appare circa a 88 anni. Le curve di questi paesi sono caratterizzate dalla quasi assenza di mortalità infantile, accidentale e prematura, concentrando tutta la massa di probabilità sulla componente legata alla morte per invecchiamento. Il secondo gruppo appare più eterogeneo ma comprende tutte curve che presentano in particolare livelli di mortalità prematura più elevati rispetto alle curve del primo gruppo e in generale uno spostamento della curva verso età inferiori che si riflettono in una aspettativa di vita più corta.

La Figura A.4 e la Tabella 4.5 non mostrano particolari indicazioni contrarie alla convergenza delle catene.

4.5.2 Applicazione modello basato sul *FPSBP*

I due gruppi identificati dal modello basato sul *FDP* non appaiono distinti solamente da un punto di vista di forme funzionali ma appaiono anche ben separati spazialmente. Rappresentando sulla mappa il raggruppamento



(a) Uniforme

(b) Modello di Siler

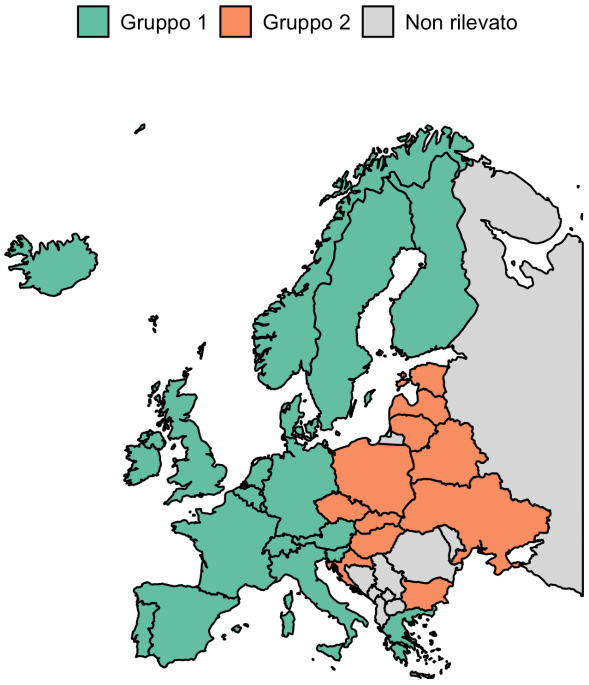
(c) Nessuna allocazione nei gruppi vuoti

(d) Allocazioni nei gruppi prima vuoti

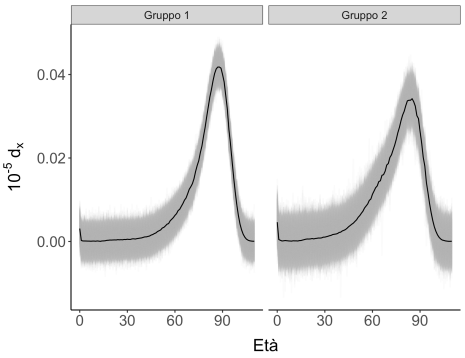
Figura 4.6: Risultati dello studio sulla robustezza della distribuzione a priori degli atomi funzionali Θ_h.

Tabella 4.5: Mortalità in Europa - Controllo convergenza modello basato sul *FDP* per il parametro di varianza complessiva σ^2, il parametro di concentrazione α e il parametro κ che regola la funzione di covarianza esponenziale in cui si è fissato il parametro relativo alla magnitudine.

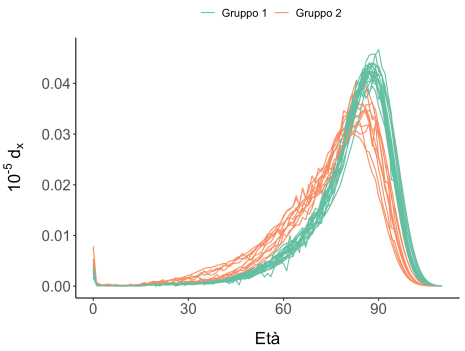
Diagnostica	σ^2	α	κ
Effective sample size (su 10 000)	8 623	4 215	2 752
Potential scale reduction factor	1	1	1



(a) Disposizione spaziale del raggruppamento ottenuto



(b) Atomi funzionali estratti



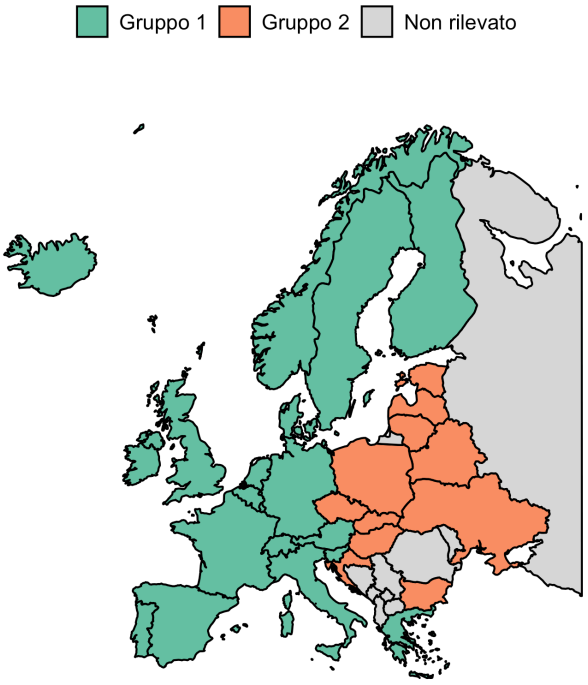
(c) Raggruppamento ottenuto

Figura 4.7: Risultati del modello basato sul *FDP* sulle curve di mortalità europee del 2013.

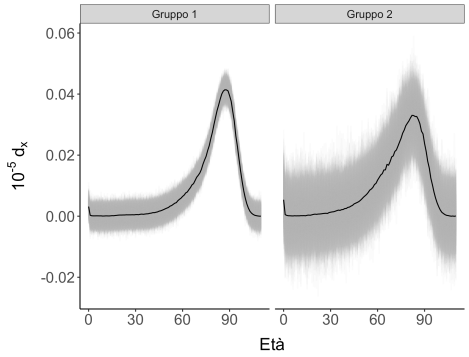
ottenuto si osserva che gli stati che presentano curve di mortalità con caratteristiche simili si trovano geograficamente vicini nello spazio. Siccome in generale non risulta ragionevole ipotizzare uno specifico *trend* spaziale valido su ampia scala ma si vuole solo tener conto della vicinanza e della dipendenza fra le osservazioni, si adatta a queste curve un modello basato sul *FPSBP* in cui il modello *probit* alla base dei *random weights* ha solo due parametri, uno di intercetta ed uno relativo alla matrice di covarianza con struttura esponenziale. In questo modo si va a modellare una superficie liscia sopra le osservazioni che favorisce l'allocazione allo stesso gruppo delle osservazioni vicine spazialmente.

I risultati di questo modello sono rappresentati in Figura 4.8 e sono gli stessi di quelli ottenuti in precedenza con l'unica differenza che l'atomo funzionale del secondo gruppo presenta secondo questo modello una variabilità maggiore. Le Figure A.5 e A.6 e la Tabella 4.6 non mostrano particolari indicazioni contrarie alla convergenza delle catene.

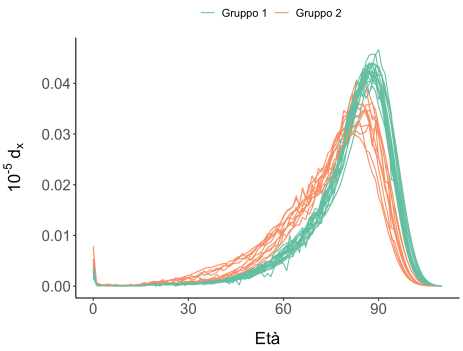
Anche in questo caso l'applicazione del modello basato sul *FPSBP* ha portato a risultati molto simili a quelli ottenuti senza la specificazione del modello *probit*. A differenza però di quanto mostrato nel caso precedente, in questo caso la distribuzione dei *random weights* con parametri aggiornati non appare nemmeno così informativa e l'allocazione è guidata quasi esclusivamente dalla verosimiglianza che dipende dalla forma funzionale delle osservazioni. Il fatto che la priori non risulti molto informativa può essere dovuto al modo in cui è stata calcolata la matrice di distanze fra le osservazioni e al fatto che sono presenti anche stati molto distanti ma con caratteristiche in termini di mortalità simili, e stati vicini ma con caratteristiche molto differenti. Una possibilità per migliorare il modello può essere quella della definizione di una matrice di distanze *ad hoc* fra gli stati, che tenga conto non solo della distanza geografica ma anche di fattori socio-culturali, storici ed economici che possono rendere uno stato più o meno connesso, e dunque vicino, ad un altro.



(a) Disposizione spaziale del raggruppamento ottenuto



(b) Atomi funzionali estratti



(c) Raggruppamento ottenuto

Figura 4.8: Risultati del modello basato sul *FPSBP* sulle curve di mortalità europee del 2013.

Tabella 4.6: Mortalità in Europa - Controllo convergenza modello basato sul *FPSBP* per il parametro di varianza complessiva σ^2, il parametro κ che regola la funzione di covarianza esponenziale in cui si è fissato il parametro relativo alla magnitudine e i parametri β_{0h} e κ_h^* del modello *probit* dei quali si riporta un valore medio della diagnostica.

Diagnostica	σ^2	κ	β_0	κ_h^*
Effective sample size (su 10 000)	5 948	2 736	2 465	2 814
Potential scale reduction factor	1	1	1.09	1.05

Conclusioni

L'obiettivo di questo lavoro è stato quello di introdurre l'analisi dei dati funzionali e la costruzione di modelli bayesiani non parametrici attraverso un'applicazione in ambito demografico. In particolare si è voluta analizzare l'evoluzione spaziale e temporale della distribuzione dei decessi per età, utilizzata dai demografi come indice di progresso e salute di una popolazione.

I modelli bayesiani non parametrici si sono dimostrati per tale scopo strumenti estremamente flessibili e adatti anche ad altre tipologie di dati. Qualora si fosse in presenza di dati di diversa natura risulterebbe sufficiente cambiare la distribuzione di base o la media a priori del processo gaussiano lasciando invariata la struttura generale del modello e dell'algoritmo di stima. Nella specificazione dei due modelli presentati infatti non si sono sfruttate le informazioni sulla caratteristica forma funzionale dei dati analizzati se non appunto nella definizione di una sensata media della distribuzione a priori degli atomi funzionali.

Si è visto come i modelli bayesiani non parametrici inducano una forma di raggruppamento delle osservazioni che presenta diversi vantaggi, fra cui quello di non dover ricorrere ad un lisciamento preliminare delle osservazioni e non dover definire a priori il numero di gruppi che si vuole ottenere. Con questi modelli risulta sufficiente specificare un limite superiore a questo numero, che è molto più semplice da definire, lasciando che siano i dati stessi ad informare sul numero di gruppi in essi presenti. Questa caratteristica ha spinto l'utilizzo di questi modelli in molti contesti applicativi.

Si evidenzia infine che la modellazione delle curve tramite modelli basati sul processo di Dirichlet funzionale e il processo *Probit Stick-Breaking* funzionale

ha portato agli stessi risultati. Questo dipende probabilmente dal fatto che nel contesto analizzato i dati presentano forti e chiare differenze in termini di forma funzionale ben visibili e dunque facilmente catturabili dal modello. L'introduzione del modello *probit* alla base della costruzione dei *random weights* ha comunque permesso nello studio dell'evoluzione temporale di specificare una distribuzione a priori che tenesse conto dell'informazione aggiuntiva che il collocamento temporale fornisce sul raggruppamento.

Estensioni di questo lavoro possono riguardare l'applicazione di questi modelli in contesti in cui l'informazione proveniente dai dati è più rumorosa, ad esempio nello studio della distribuzione dei decessi per età dei singoli comuni. In questo caso la forma funzionale della curva riflette la natura prettamente discreta dei conteggi dei decessi e l'utilizzo di distribuzioni a priori più informative anche per i *random weights* può risultare utile al fine di ottenere un miglior raggruppamento funzionale.

Appendice A

Grafici delle traiettorie dei vari modelli

I grafici delle traiettorie, in inglese *traceplot*, mostrano le oscillazioni dei valori generati per ciascun parametro dall'algoritmo *MCMC* e rappresentano una diagnostica a livello grafico per la valutazione della convergenza. Indicazioni contrarie alla convergenza sono una lenta oscillazione dei valori della catena, la non stazionarietà o l'identificazione di evidenti *pattern*. A tale grafico può essere sovrapposto anche il grafico della media cumulata, il quale dopo il periodo di riscaldamento dovrebbe stabilizzarsi (Robert e Casella, 2010).

Osservando i vari grafici non sembra esserci evidenza contro la convergenza di nessuna catena generata.

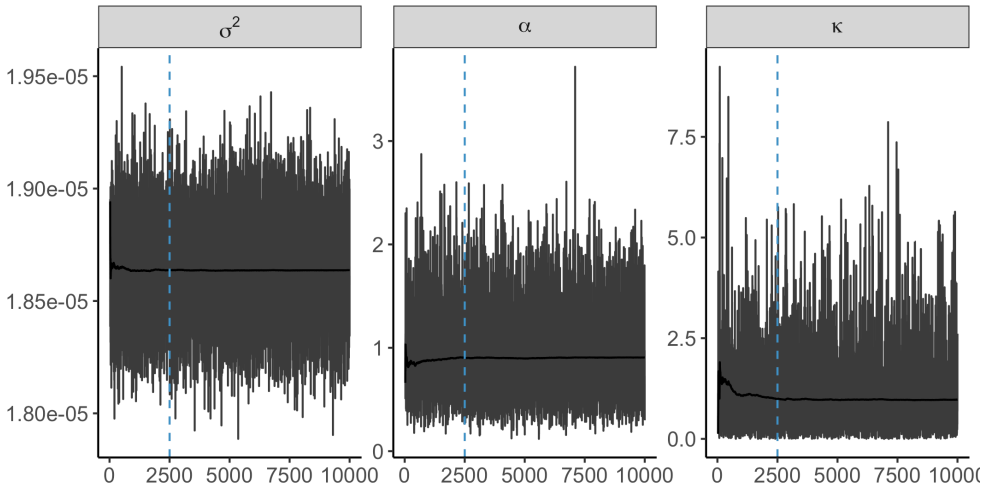


Figura A.1: Grafici delle traiettorie del modello per le curve di mortalità italiane basato sul *FDP*.

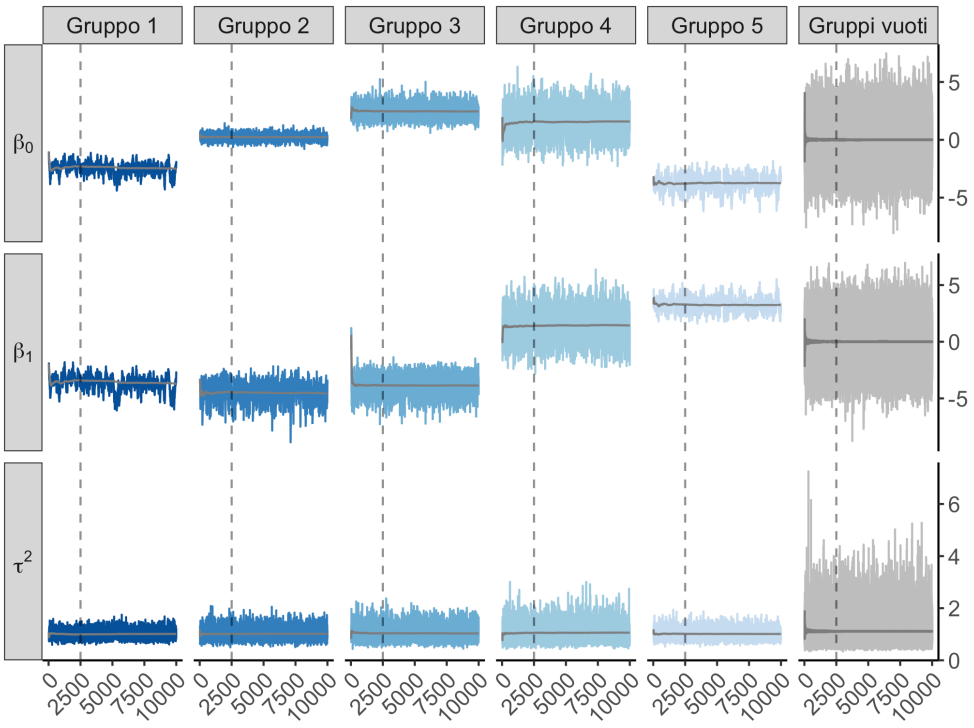


Figura A.2: Grafici delle traiettorie del modello per le curve di mortalità italiane basato sul *FPSBP* dei parametri $\beta_h = (\beta_{0h}, \beta_{1h})$ e τ_h^2 alla base del modello *probit* per ciascun *cluster*. Le traiettorie dei parametri dei gruppi vuoti sono rappresentate nella colonna a destra sovrapposte.

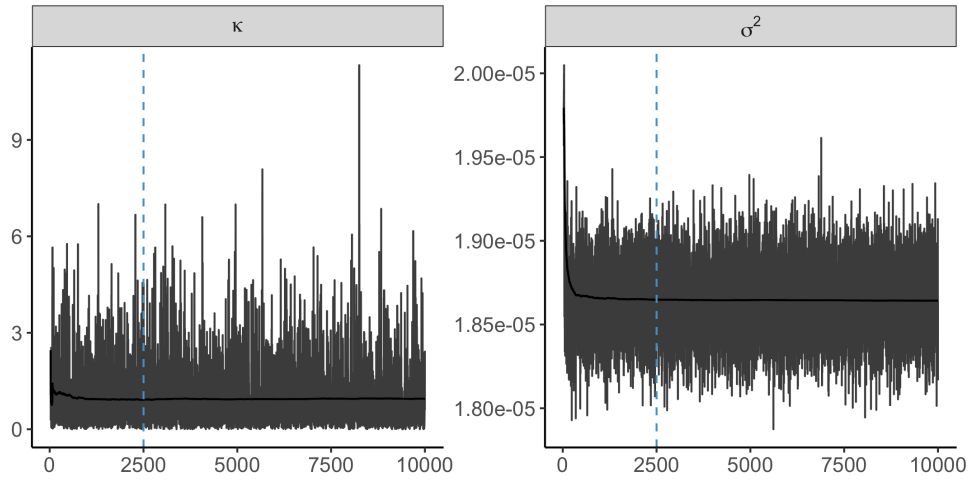


Figura A.3: Grafici delle traiettorie del modello per le curve di mortalità italiane basato sul *FPSBP* dei parametri κ della funzione di covarianza esponenziale e del parametro di varianza complessiva σ^2.

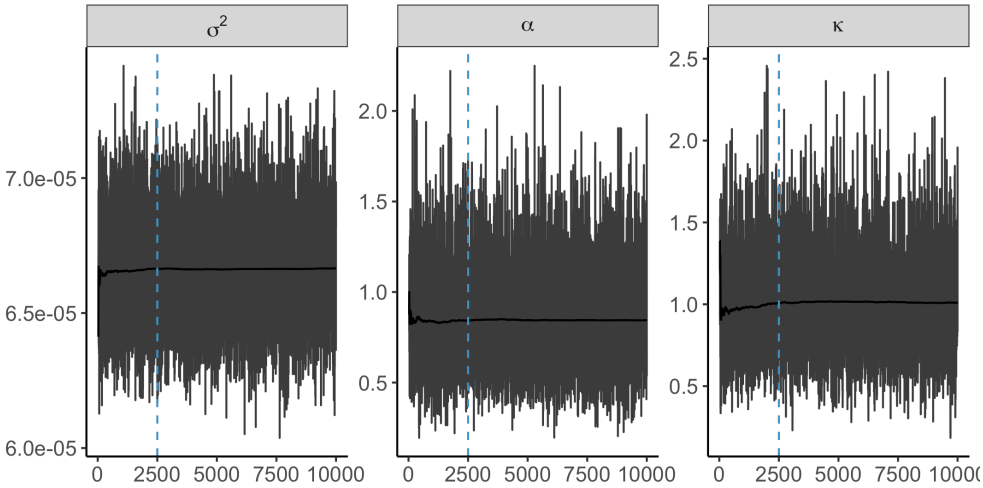


Figura A.4: Grafici delle traiettorie del modello per le curve di mortalità europee del 2013 basato sul *FDP*.

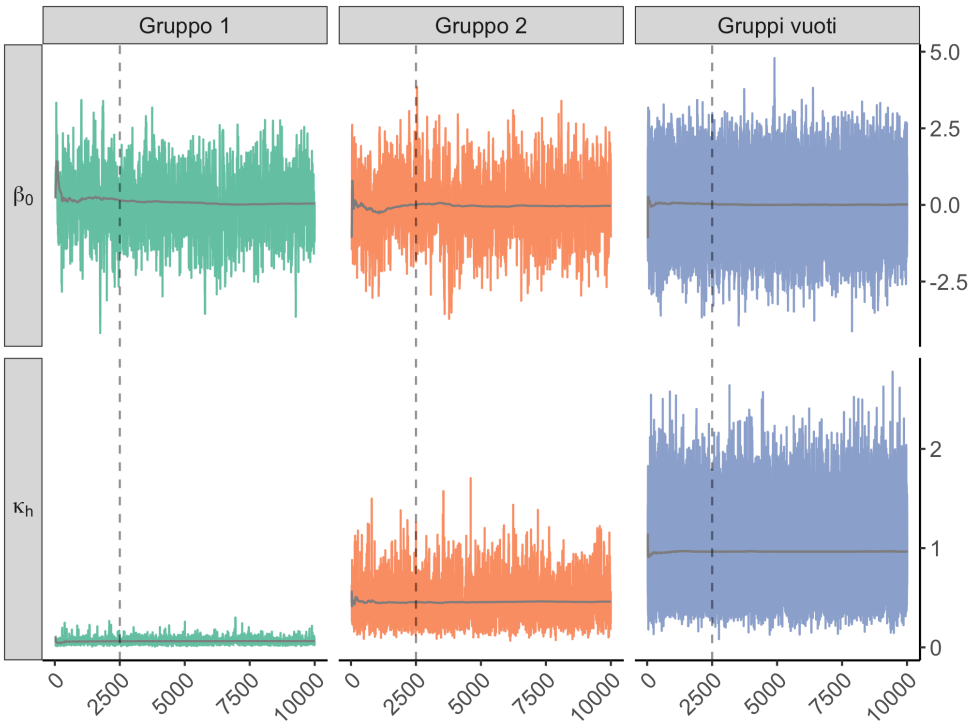


Figura A.5: Grafici delle traiettorie del modello per le curve di mortalità europee del 2013 basato sul *FPSBP* dei parametri β_{0h} e κ_h alla base del modello *probit* per ciascun *cluster*. Le traiettorie dei parametri dei gruppi vuoti sono rappresentate nella colonna a destra sovrapposte.

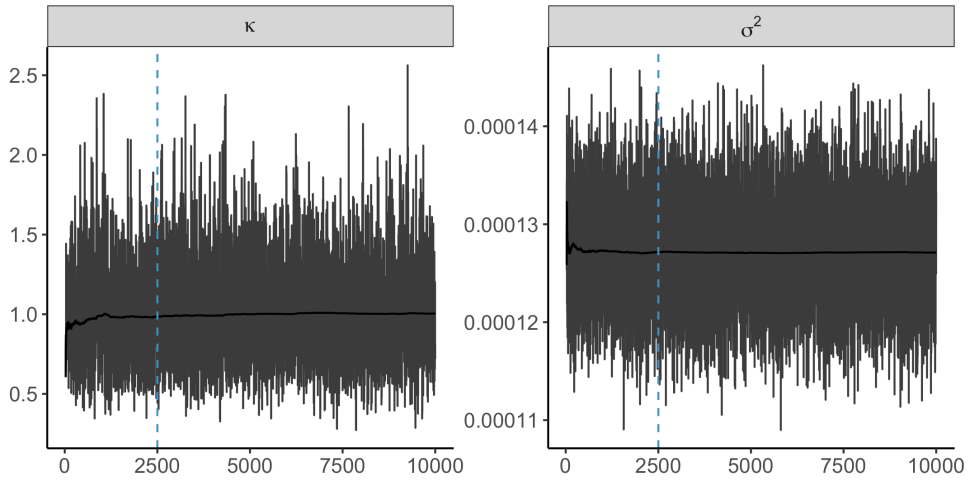


Figura A.6: Grafici delle traiettorie del modello per le curve di mortalità europee del 2013 basato sul *FPSBP* dei parametri κ della funzione di covarianza esponenziale e del parametro di varianza complessiva σ^2.

Appendice B

Codice R

B.1 Modello basato sul *FDP*

Di seguito viene proposta una possibile implementazione per la stima del modello basato sul processo di Dirichlet funzionale utilizzando l'algoritmo *Blocked Gibbs sampler*.

```
FDP = function(data, mu0, start_kappa, eps_kappa, hyper_kappa, hyper_alpha, R, N, seed){
  "
  Input:
  ------
  data = matrice dei dati di dimensione n x T (una curva per riga)
  mu0 = vettore T x 1 con la media a priori del GP
  start_kappa = vettore 2 x 1 con i punti di partenza per il Metropolis su kappa
  eps_kappa = vettore 2 x 1 con gli epsilon per regolare il Metropolis sul kappa
  hyper_kappa = matrice 2 x 2 con gli iperparametri della a priori per kappa1 e kappa2
  hyper_alpha = vettore 2 x 1 con iperparametri della a priori per alpha ~ Ga(a, b)
  R = numero di iterazioni
  N = troncamento al numero di gruppi
  seed = seme per replicare i risultati

  Output:
  -------
  lista contenente i dati, informazioni sul clustering ottenuto, parametri degli
  atomi funzionali individuati, varianza del processo
  "

  # carico librerie
  library(mvtnorm)
  library(Hmisc)
  library(fields)
```

```
# 1) Aggiornamento S_i ---------------------------------------
cluster.ind = function(dati, stickbreaking.v, theta, sigma2, N){
  stickbreaking.v.comp = cumprod(1-stickbreaking.v)
  stickbreaking.w = rep(0,N)
  stickbreaking.w[1] = stickbreaking.v[1]
  stickbreaking.w[2:N] = sapply(2:N, function(h) stickbreaking.v[h]*
                                stickbreaking.v.comp[h-1])
  s = rep(0,nrow(dati))
  logL = matrix(0, nrow = nrow(dati), ncol = N)
  stickbreaking.p = matrix(0, nrow = nrow(dati), ncol = N)
  for(i in 1:nrow(dati)){
    for(h in 1:N){
      logL[i,h] = sum(dnorm(dati[i,], mean = theta[h,], sd = sqrt(sigma2), log = T))
      stickbreaking.p[i,h] = stickbreaking.w[h]*exp(logL[i,h])
    }
    stickbreaking.p = stickbreaking.p / apply(stickbreaking.p, 1, sum)
  }
  s = as.vector(rMultinom(stickbreaking.p, 1))
  return(list(s = as.numeric(factor(s)), stickbreaking.p = stickbreaking.p))
}

# 2) Aggiornamento v_h ---------------------------------------
stickbreak.update = function(s, alpha, N){
  stickbreaking.v = rep(NA, N)
  stickbreaking.v[1:(N-1)] = sapply(1:(N-1), function(h)
    rbeta(1, 1+sum(s==h), alpha+sum(s>h)))
  stickbreaking.v[N] = 1
  return(stickbreaking.v)
}

# 3) Aggiornamento theta_h -----------------------------------
covariance.matrix = function(k, distance, nu){
  Cov = k[1] * exp(-k[2]*distance)
  diag(Cov) = diag(Cov) + nu*k[1]
  return(Cov)
}
functional.atoms.update = function(dati, C, s, sigma2, mu0, N, nt){
  nh = as.numeric(table(factor(s, levels = 1:N)))
  theta = matrix(NA, nrow = N, ncol = nt)
  for(h in 1:N){
    if(nh[h] > 0){
      m = colMeans(dati[s == h, , drop = F]) - mu0
      L = t(Rfast::cholesky(C + sigma2/nh[h] * diag(nt)))
      mutilde = mu0 + as.vector(C %*% backsolve(t(L), forwardsolve(L, m)))
      v = forwardsolve(L, C)
      Sigmatilde = C - t(v) %*% v
      theta[h,] = drop(mvtnorm::rmvnorm(1, mean = mutilde,
```

```
                                        sigma = Sigmatilde, method = "chol"))
    } else{
      theta[h,] = drop(mvtnorm::rmvnorm(1, mean = mu0, sigma = C, method = "chol"))
    }
  }
  return(theta)
}

# 4) aggiornamento kappa (metropolis) -------------------------
lprior.k = function(k, hyper_kappa){
  # k = log(kappa)
  sum(dgamma(exp(k), hyper_kappa[,1], hyper_kappa[,2], log = T)) + sum(k)
}

loglikelihood.GP = function(theta, k, distance, mu0, dati, s, nu, N){
  # k = kappa
  nh = as.numeric(table(factor(s, levels = 1:N)))
  C = covariance.matrix(k, distance, nu)
  C_chol = t(Rfast::cholesky(C))
  llik = 0
  for(h in 1:N){
    if(nh[h] > 0){
      m = colMeans(dati[s == h, , drop = F]) - mu0
      L = t(Rfast::cholesky(C + sigma2/nh[h] * diag(nt)))
      mutilde = mu0 + as.vector(C %*% backsolve(t(L), forwardsolve(L, m)))
      v = forwardsolve(L, C)
      Sigmatilde = C - t(v) %*% v
      Sigma_chol = t(Rfast::cholesky(Sigmatilde))
      llik = llik - sum(log(diag(Sigma_chol))) -
        0.5*(t(theta[h,] - mutilde) %*%
               backsolve(t(Sigma_chol), forwardsolve(Sigma_chol, theta[h,]-mutilde)))
    } else{
      llik = llik - sum(log(diag(C_chol))) -
        0.5*(t(theta[h,] - mu0) %*%
               backsolve(t(C_chol), forwardsolve(C_chol, theta[h,] - mu0)))
    }
  }
  return(llik)
}

lposterior.k = function(k, hyper_kappa, theta, distance, mu0, dati, s, nu, N){
  lprior.k(k, hyper_kappa) +
    loglikelihood.GP(theta, exp(k), distance, mu0, dati, s, nu, N)
}

metropolis.k = function(k0, eps, hyper_kappa, theta, distance, mu0, dati, s, nu, N){
  accepted = numeric(2)
  k = log(k0)
  for(j in 1:2){
```

```
    kstar = k
    kstar[j] = k[j] + runif(1, -eps[j], eps[j])
    alp = min(1, exp(lposterior.k(kstar,hyper_kappa,theta,distance,mu0,dati,s,nu,N) -
                     lposterior.k(k,hyper_kappa,theta,distance,mu0,dati,s,nu,N)))
    if(runif(1) < alp){
      k[j] = kstar[j]
      accepted[j] = 1
    }
  }
  return(list(k = exp(k), accepted = accepted))
}

# 5) Aggiornamento sigma2 -------------------------------------
complex.variance = function(s, hyper, theta, dati){
  resid = dati - theta[s,]
  nT = nrow(dati) * ncol(dati)
  sigma2 = 1 / rgamma(1, hyper[1] + nT/2, hyper[2] + 0.5*sum(resid^2))
}

# 6) Aggiornamento alpha --------------------------------------
concentration.alpha = function(stickbreaking.v, hyper){
  N = length(stickbreaking.v)
  alpha = rgamma(1, hyper[1]+N-1, hyper[2]-cumsum(log(1-stickbreaking.v))[N-1])
  return(alpha)
}


# ALGORITMO DI STIMA ------------------------------------------

# imposta il seme
set.seed(seed)

# trasforma le curve in distribuzioni che sommano ad 1
data = data / apply(data, 1, sum)

# inizializzazioni
punti = as.numeric(colnames(data))
nt = length(punti)
distance2 = fields::rdist(punti/10^6)^2
v.stick = c(rep(0.5, N-1), 1)
theta = matrix(0, nrow = N, ncol = nt)
k.info = list(k = start_kappa, accepted = c(0,0))
sigma2 = 100
alpha = 5
nu = 0.0001
hyper_sigma = c(0.1, 0.1)

# allocazione della memoria
cluster.out = matrix(0, nrow = R, ncol = nrow(data))
```

```
  stickbreaking.p.out = array(0, dim = c(R, nrow(data), N))
  stickbreaking.v.out = matrix(0, nrow = R, ncol = N)
  theta.out = array(0, dim = c(R, N, nt))
  k.out = matrix(0, nrow = R, ncol = 2)
  accepted.out = matrix(0, nrow = R, ncol = 2)
  sigma2.out = rep(0, R)
  alpha.out = rep(0, R)
  number.cluster = rep(0, R)

  pb = txtProgressBar(0, R, style = 3)   # mostra avanzamento
  for(r in 1:R){
    cluster = cluster.ind(data, v.stick, theta, sigma2, N)
    v.stick = stickbreak.update(cluster$s, alpha, N)
    Cov.matrix = covariance.matrix(k.info$k, distance2, nu)
    theta = functional.atoms.update(data, Cov.matrix, cluster$s, sigma2, mu0, N, nt)
    k.info = metropolis.k(k.info$k, eps_kappa, hyper_kappa, theta, distance2, mu0,
                          data, cluster$s, nu, N)
    sigma2 = complex.variance(cluster$s, hyper_sigma, theta, data)
    alpha = concentration.alpha(v.stick, hyper_alpha)

    # salvataggio dei risultati
    cluster.out[r,] = cluster$s
    number.cluster[r] = length(unique(cluster$s))
    stickbreaking.p.out[r,,] = cluster$stickbreaking.p
    stickbreaking.v.out[r,] = v.stick
    theta.out[r,,] = theta
    k.out[r,] = k.info$k
    accepted.out[r,] = k.info$accepted
    sigma2.out[r] = sigma2
    alpha.out[r] = alpha

    setTxtProgressBar(pb, r)  # aggiorna stato avanzamento
  }
  close(pb)  # chiude stato aggiornamento

  out = list(data = data, cluster = cluster.out, num_cluster = number.cluster,
             p_stick = stickbreaking.p.out, v_stick = stickbreaking.v.out,
             theta = theta.out, kappa = k.out, kappa_accepted = accepted.out,
             sigma2 = sigma2.out, alpha = alpha.out, seed = seed)

  return(out)
}
```

B.2 Modello basato sul *FPSBP*

Di seguito viene proposta una possibile implementazione per la stima del modello basato sul processo *Probit Stick Breaking* funzionale utilizzando

l'algoritmo *Blocked Gibbs sampler*. Nella seguente implementazione si è ipotizzata una specifica struttura del modello *probit* che corrisponde a quella specificata per la modellazione dell'evoluzione temporale della curva, in particolare si assume che $\alpha_h \sim N_n(X\beta_h, \tau_h^2 I_n)$.

```
FPSBP = function(data, X, mu0, start_kappa, eps_kappa, hyper_kappa, R, N, seed){
  "
  Input:
  ------
  data = matrice dei dati di dimensione n x T (ogni riga è una curva)
  X = matrice di disegno per il modello probit di dimensione n x p
  mu0 = vettore T x 1 con la media a priori del GP
  start_kappa = vettore 2 x 1 con i punti di partenza per il Metropolis su kappa
  eps_kappa = vettore 2 x 1 con gli epsilon per regolare il Metropolis su kappa
  hyper_kappa = matrice 2 x 2 con gli iperparametri della a priori per kappa1 e kappa2
  hyper_alpha = vettore 2 x 1 con iperparametri della a priori per alpha ~ Ga(a, b)
  R = numero di iterazioni
  N = troncamento al numero di gruppi
  seed = seme per replicare i risultati

  Output:
  -------
  lista contenente i dati e la matrice di disegno utilizzati, informazioni sul clustering
  ottenuto, quantità del modello probit alla base dei random weights, parametri degli
  atomi funzionali individuati, varianza del processo
  "

  # carico librerie
  library(mvtnorm)
  library(Hmisc)
  library(fields)
  library(truncnorm)

  # 1) Aggiornamento S_i ----------------------------------------------------
  cluster_update = function(X, beta, data, theta, sigma2){
    N = nrow(theta)
    n = nrow(data)
    p = ncol(X)
    v_stick = pnorm(X %*% beta[1:p,])
    v_stick[,N] = 1
    v_stick_comp = t(apply(1-v_stick, 1, cumprod))
    w_stick = matrix(NA, nrow = n, ncol = N)
    w_stick[,1] = v_stick[,1]
    w_stick[,2:N] = sapply(2:N, function(h) v_stick[,h]*v_stick_comp[,h-1])
    p_stick = matrix(NA, nrow = n, ncol = N)
    for(i in 1:n){
      for(h in 1:N){
        logL = sum(dnorm(data[i,], mean = theta[h,], sd = sqrt(sigma2), log = T))
```

```
      p_stick[i,h] = w_stick[i,h]*exp(logL)
    }
  }
  p_stick[apply(p_stick, 1, sum) == 0,] = w_stick[apply(p_stick, 1, sum) == 0,]
  p_stick = p_stick / apply(p_stick, 1, sum)
  s = as.vector(Hmisc::rMultinom(probs = p_stick, m = 1))
  return(list(s = as.numeric(factor(s)), v_stick = v_stick,
              w_stick = w_stick, p_stick = p_stick))
}

# 2) Aggiornamento alpha_h ------------------------------------------------
alpha_update = function(alpha_stick, s, X, beta){
  n = nrow(X)
  p = ncol(X)
  N = ncol(beta)
  alpha_stick_new = z_stick = matrix(0, nrow = n, ncol = N)

  for(i in 1:n){
    for(h in 1:N){
      if(h > s[i]) break
      lower = ifelse(h < s[i], -Inf, 0)
      upper = ifelse(h < s[i], 0, Inf)
      z_stick[i,h] = rtruncnorm(1, a=lower, b=upper, mean=alpha_stick[i,h], sd=1)
      mu_post = (X[i,,drop=F] %*% beta[1:p,h] + z_stick[i,h]*beta[p+1,h]) / (1+beta[p+1,h])
      sd_post = sqrt(beta[p+1,h] / (1+beta[p+1,h]))
      alpha_stick_new[i,h] = rtruncnorm(1, a=lower, b=upper, mean=mu_post, sd=sd_post)
    }
  }
  return(alpha_stick_new)
}

# 3) Aggiornamento beta_h -------------------------------------------------
beta_update = function(alpha_stick, X, s){
  p = ncol(X)
  N = ncol(alpha_stick)
  beta_matrix = matrix(NA, p+1, N)
  sigma2_int = rep(NA, N)
  a_tau = b_tau = 10
  mu_beta = rep(0, p)
  sigma0 = 2

  for(h in 1:N){
    alpha_stick = alpha_stick[s >= h, , drop = F]
    X = X[s >= h, , drop = F]
    s = s[s >= h]
    n = nrow(X)
    a_post = a_tau + n/2
    q1 = solve(t(X) %*% X + 1/sigma0*diag(p))
```

```
    mu_post = q1 %*% (1/sigma0*diag(p) %*% mu_beta + t(X) %*% alpha_stick[,h])
    b_post = b_tau + 0.5 * (t(alpha_stick[,h]) %*% alpha_stick[,h] + t(mu_beta) %*%
                            (1/sigma0 * diag(p)) %*% mu_beta -
                            t(mu_post) %*% solve(q1) %*% mu_post)
    tau2 = 1/rgamma(1, shape = a_post, rate = b_post)
    Sigma_post = tau2 * q1
    beta = drop(mvtnorm::rmvnorm(1, mean=mu_post, sigma=Sigma_post, method="chol"))
    beta_matrix[,h] = c(beta, tau2)
  }
  return(beta_matrix)
}

# 4) Aggiornamento theta_h ------------------------------------------------
covariance = function(kappa, distance, zeta){
  Cov = kappa[1] * exp(-kappa[2]*distance)
  diag(Cov) = diag(Cov) + zeta*kappa[1]
  return(Cov)
}
functional_atoms_update = function(data, s, mu0, K, sigma2, N){
  nh = as.numeric(table(factor(s, levels = 1:N)))
  nt = ncol(data)
  theta = matrix(NA, nrow = N, ncol = nt)
  for(h in 1:N){
    if(nh[h] > 0){
      m = colMeans(data[s == h, , drop = F]) - mu0
      L = t(Rfast::cholesky(K + sigma2/nh[h] * diag(nt)))
      mutilde = mu0 + as.vector(K %*% backsolve(t(L), forwardsolve(L, m)))
      v = forwardsolve(L, K)
      Sigmatilde = K - t(v) %*% v
      theta[h,] = drop(mvtnorm::rmvnorm(1, mean = mutilde,
                                        sigma = Sigmatilde, method = "chol"))
    } else{
      theta[h,] = drop(mvtnorm::rmvnorm(1, mean = mu0, sigma = K/10^2, method = "chol"))
    }
  }
  return(theta)
}


# 5) Aggiornamento kappa (metropolis) -------------------------------------
lprior = function(par, hyper){
  # par = log(kappa)
  sum(dgamma(exp(par), hyper[,1], hyper[,2], log = T)) + sum(par)
}
llik = function(par, theta, distance, mu0, data, s, zeta){
  # k = kappa
  N = nrow(theta)
  nh = as.numeric(table(factor(s, levels = 1:N)))
  C = covariance(par, distance, zeta)
```

```
  C_chol = t(Rfast::cholesky(C))
  llik = 0
  for(h in 1:N){
    if(nh[h] > 0){
      m = colMeans(data[s == h, , drop = F]) - mu0
      L = t(Rfast::cholesky(C + sigma2/nh[h] * diag(nt)))
      mutilde = mu0 + as.vector(C %*% backsolve(t(L), forwardsolve(L, m)))
      v = forwardsolve(L, C)
      Sigmatilde = C - t(v) %*% v
      Sigma_chol = t(Rfast::cholesky(Sigmatilde))
      llik = llik - sum(log(diag(Sigma_chol))) - 0.5*(t(theta[h,] - mutilde) %*%
             backsolve(t(Sigma_chol), forwardsolve(Sigma_chol, theta[h,] - mutilde)))
    } else{
      llik = llik - sum(log(diag(C_chol))) - 0.5*(t(theta[h,] - mu0) %*%
             backsolve(t(C_chol), forwardsolve(C_chol, theta[h,] - mu0)))
    }
  }
  return(llik)
}
lposterior = function(par, hyper, theta, distance, mu0, data, s, zeta){
  # par = log(kappa)
  # priori per log(kappa)
  # verosimiglianza per kappa calcolata in exp(par) per invarianza
  lprior(par, hyper) + llik(exp(par), theta, distance, mu0, data, s, zeta)
}
metropolis_kappa = function(start, eps, hyper, theta, distance, mu0, data, s, zeta){
  d = length(start)
  accepted = numeric(d)
  k = log(start)   # metropolis sulla trasformata logaritmica di kappa
  for(j in 1:d){
    kstar = k
    kstar[j] = k[j] + runif(1, -eps[j], eps[j])
    alp = min(1, exp(lposterior(kstar, hyper, theta, distance, mu0, data, s, zeta) -
                     lposterior(k, hyper, theta, distance, mu0, data, s, zeta)))
    if(runif(1) < alp){
      k[j] = kstar[j]
      accepted[j] = 1
    }
  }
  return(list(k = exp(k), accepted = accepted))  # ritorna il vettore kappa > 0
}

# 6) Aggiornamento sigma2 ------------------------------------------------
sigma2_update = function(data, s, theta, hyper){
  resid = data - theta[s,]
  sigma2 = 1 / rgamma(1, shape = hyper[1] + nrow(data)*ncol(data)/2,
                      rate = hyper[2] + 0.5*sum(resid^2))
  return(sigma2)
}
```

```
# ALGORITMO DI STIMA ------------------------------------------------------

# imposta il seme
set.seed(seed)

# trasforma le curve in distribuzioni che sommano ad 1 al posto di 100 mila
data = data / apply(data, 1, sum)

# standardizza matrice di disegno del modello probit
X[,-1] = scale(X[,-1])
n = nrow(X)
p = ncol(X)

# inizializzazioni
punti = as.numeric(colnames(data))
nt = length(punti)
distance2 = fields::rdist(punti/10^6)^2
sigma2 = 100
hyper_sigma2 = rep(10^-1, 2)
theta = matrix(0, nrow = N, ncol = nt)
kappa_mcmc = list(k = start_kappa, accepted = c(0,0))
zeta = 10^-4
beta = matrix(rep(c(rep(0,p), 1), N), nrow = p+1, ncol = N)
alpha_stick = matrix(0, nrow = n, ncol = N) +
  matrix(runif(n*N, min = -0.25, max = 0.25), nrow = n, ncol = N)

# allocazione della memoria
cluster_out = matrix(0, nrow = R, ncol = n)
number_cluster_out = rep(0, R)
v_stick_out = array(0, dim = c(R, n, N))
w_stick_out = array(0, dim = c(R, n, N))
p_stick_out = array(0, dim = c(R, n, N))
alpha_stick_out = array(0, dim = c(R, n, N))
beta_out = array(0, dim = c(R, p+1, N))
theta_out = array(0, dim = c(R, N, nt))
kappa_out = matrix(0, nrow = R, ncol = 2)
kappa_accepted_out = matrix(0, nrow = R, ncol = 2)
sigma2_out = rep(0, R)

pb = txtProgressBar(0, R, style = 3)   # per visualizzare stato di avanzamento
for(r in 1:R){
  cluster = cluster_update(X, beta, data, theta, sigma2)
  alpha_stick = alpha_update(alpha_stick, cluster$s, X, beta)
  beta = beta_update(alpha_stick, X, cluster$s)
  K = covariance(kappa_mcmc$k, distance2, zeta)
  theta = functional_atoms_update(data, cluster$s, mu0, K, sigma2, N)
  kappa_mcmc = metropolis_kappa(kappa_mcmc$k, eps_kappa, hyper_kappa, theta,
```

```
                                distance2, mu0, data, cluster$s, zeta)
    sigma2 = sigma2_update(data, cluster$s, theta, hyper_sigma2)

    # salvataggio dei risultati
    cluster_out[r,] = cluster$s
    number_cluster_out[r] = n_distinct(cluster$s)
    v_stick_out[r,,] = cluster$v_stick
    w_stick_out[r,,] = cluster$w_stick
    p_stick_out[r,,] = cluster$p_stick
    alpha_stick_out[r,,] = alpha_stick
    beta_out[r,,] = beta
    theta_out[r,,] = theta
    kappa_out[r,] = kappa_mcmc$k
    kappa_accepted_out[r,] = kappa_mcmc$accepted
    sigma2_out[r] = sigma2

    setTxtProgressBar(pb, r)  # aggiorna stato avanzamento
  }
  close(pb)  # chiude stato aggiornamento
  colnames(beta_out) = c(colnames(X), "tau2")

  out = list(data = data, covariates = X,
             cluster = cluster_out, num_cluster = number_cluster_out,
             v_stick = v_stick_out, w_stick = w_stick_out, p_stick = p_stick_out,
             alpha_stick = alpha_stick_out, beta = beta_out,
             theta = theta_out, kappa = kappa_out, kappa_accepted = kappa_accepted_out,
             sigma2 = sigma2_out, seed = seed)
  return(out)
}
```

Bibliografia

Aliverti, Emanuele, Stefano Mazzuco e Bruno Scarpa. «Dynamic modeling of mortality via mixtures of skewed distribution functions». In: (feb. 2021).

Antoniak, Charles E. «Mixtures of Dirichlet processes with applications to Bayesian nonparametric problems». In: *The annals of statistics* (1974), pp. 1152–1174.

Azzalini, Adelchi. «A class of distributions which includes the normal ones». In: *Scandinavian journal of statistics* (1985), pp. 171–178.

Azzalini, Adelchi e Bruno Scarpa. *Analisi dei dati e data mining*. Springer Science & Business Media, 2009.

Barbieri, Magali et al. «Data resource profile: the human mortality database (HMD)». In: *International journal of epidemiology* 44.5 (2015), pp. 1549–1556.

Basu, Sanjib e Siddhartha Chib. «Marginal likelihood and Bayes factors for Dirichlet process mixture models». In: *Journal of the American Statistical Association* 98.461 (2003), pp. 224–235.

Berger, James O e Luis R Pericchi. «The intrinsic Bayes factor for model selection and prediction». In: *Journal of the American Statistical Association* 91.433 (1996), pp. 109–122.

Blackwell, David e James B MacQueen. «Ferguson distributions via Pólya urn schemes». In: *The annals of statistics* 1.2 (1973), pp. 353–355.

Chung, Yeonseung e David B Dunson. «Nonparametric Bayes conditional distribution modeling with variable selection». In: *Journal of the American Statistical Association* 104.488 (2009), pp. 1646–1660.

Denison, David GT et al. *Bayesian methods for nonlinear classification and regression*. Vol. 386. John Wiley & Sons, 2002.

Dunson, David B e Amy H Herring. «Semiparametric Bayesian latent trajectory models». In: *Proceedings ISDS Discussion Paper* 16 (2006).

Durante, Daniele. «Conjugate Bayes for probit regression via unified skew-normal distributions». In: *Biometrika* 106.4 (ago. 2019), pp. 765–779.

Escobar, Michael D e Mike West. «Bayesian density estimation and inference using mixtures». In: *Journal of the american statistical association* 90.430 (1995), pp. 577–588.

Friedman, Jerome, Trevor Hastie, Robert Tibshirani et al. *The elements of statistical learning*. Vol. 1. 10. Springer series in statistics New York, 2001.

Gelman, A. et al. *Bayesian data analysis*. Chapman e Hall/CRC, 2015.

Geweke, John. «Evaluating the accuracy of sampling-based approaches to the calculations of posterior moments». In: *Bayesian statistics* 4 (1992), pp. 641–649.

Gompertz, Benjamin. «On the nature of the function expressive of the law of human mortality, and on a new mode of determining the value of life contingencies». In: *Philosophical transactions of the Royal Society of London* 115 (1825), pp. 513–583.

Hjort, Nils Lid et al. *Bayesian nonparametrics*. Vol. 28. Cambridge University Press, 2010.

HMD Data Availability. `https://www.mortality.org/cgi-bin/hmd/DataAvailability.php`. Visitato il 28/02/2022.

Ishwaran, Hemant e Lancelot F James. «Gibbs sampling methods for stick-breaking priors». In: *Journal of the American Statistical Association* 96.453 (2001), pp. 161–173.

Ishwaran, Hemant e Mahmoud Zarepour. «Markov chain Monte Carlo in approximate Dirichlet and beta two-parameter process hierarchical models». In: *Biometrika* 87.2 (2000), pp. 371–390.

Jacques, Julien e Cristian Preda. «Functional data clustering: a survey». In: *Advances in Data Analysis and Classification* 8.3 (2014), pp. 231–255.

Legramanti, Sirio, Tommaso Rigon e Daniele Durante. «Bayesian testing for exogenous partition structures in stochastic block models». In: *Sankhya A* (2020), pp. 1–19.

Lexis, Wilhelm Hector Richard Albrecht. *Sur la durée normale de la vie humaine et sur la théorie de la stabilité des rapports statistiques*. Vve. F. Henry, 1879.

MacEachern, Steven N. «Estimating normal means with a conjugate style Dirichlet process prior». In: *Communications in Statistics-Simulation and Computation* 23.3 (1994), pp. 727–741.

Makeham, William Matthew. «On the law of mortality and the construction of annuity tables». In: *Journal of the Institute of Actuaries* 8.6 (1860), pp. 301–310.

Mazzuco, Stefano, Bruno Scarpa e Lucia Zanotto. «A mortality model based on a mixture distribution function». In: *Population Studies* 72.2 (2018), pp. 191–200.

Miller, Jeffrey W e Matthew T Harrison. «Mixture models with a prior on the number of components». In: *Journal of the American Statistical Association* 113.521 (2018), pp. 340–356.

Muliere, Pietro e Luca Tardella. «Approximating distributions of random functionals of Ferguson-Dirichlet priors». In: *Canadian Journal of Statistics* 26.2 (1998), pp. 283–297.

Müller, Peter et al. *Bayesian nonparametric data analysis*. Springer, 2015.

Papaspiliopoulos, Omiros e Gareth O Roberts. «Retrospective Markov chain Monte Carlo methods for Dirichlet process hierarchical models». In: *Biometrika* 95.1 (2008), pp. 169–186.

Pearson, Karl. *The chances of death, and other studies in evolution*. Vol. 2. E. Arnold, 1897.

Preston, Samuel H, Patrick Heuveline e Michel Guillot. *Demography, measuring and modeling population processes*. blackwell Publishing, 2001.

Raftery, Adrian et al. «Estimating the integrated likelihood via posterior simulation using the harmonic mean identity». In: *Bayesian statistics* 8 (gen. 2007).

Ramsay, James O, Giles Hooker e Spencer Graves. *Functional Data Analysis with R and MATLAB*. Springer, 2009.

Ramsay, James O e Bernard W Silverman. *Functional Data Analysis*. Springer, 2005.

Rasmussen, C.E. e C.K.I Williams. «Gaussian processes for machine learning». In: *International Journal of Neural Systems* 14 (2006).

Ren, Lu et al. «Logistic stick-breaking process». In: *Journal of Machine Learning Research* 12.1 (2011).

Robert, Christian P e George Casella. *Introducing monte carlo methods with r*. Vol. 18. Springer, 2010.

Rodriguez, Abel e David B Dunson. «Nonparametric Bayesian models through probit stick-breaking processes». In: *Bayesian analysis (Online)* 6.1 (2011).

Rodriguez, Abel, David B Dunson e Jack Taylor. «Bayesian hierarchically weighted finite mixture models for samples of distributions». In: *Biostatistics* 10.1 (2009), pp. 155–171.

Rodriguez, Abel e Peter Müller. «Nonparametric bayesian inference». In: *NSF-CBMS Regional Conference Series in Probability and Statistics*. Vol. 9. JSTOR. 2013, pp. i–110.

Scarpa, Bruno e David B Dunson. «Bayesian hierarchical functional data analysis via contaminated informative priors». In: *Biometrics* 65.3 (2009), pp. 772–780.

Siler, William. «A competing-risk model for animal mortality». In: *Ecology* 60.4 (1979), pp. 750–757.

— «Parameters of mortality in human populations with widely varying life spans». In: *Statistics in medicine* 2.3 (1983), pp. 373–380.

Stephens, Matthew. «Dealing with label switching in mixture models». In: *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 62.4 (2000), pp. 795–809.

Thompson, Wesley K e Ori Rosen. «A Bayesian model for sparse functional data». In: *Biometrics* 64.1 (2008), pp. 54–63.

Walker, Stephen G. «Sampling the Dirichlet mixture model with slices». In: *Communications in Statistics—Simulation and Computation* 36.1 (2007), pp. 45–54.

Wilmoth, JR et al. *Methods protocol for the human mortality database*. 2019.

Zanotto, Lucia, Vladimir Canudas-Romo e Stefano Mazzuco. «A Mixture-Function Mortality Model: Illustration of the Evolution of Premature Mortality». In: *European Journal of Population* 37 (mar. 2020).